Statistical Image Analysis in Cone-beam Computed Tomography

Doshi, Susan

Award date:
2014

Awarding institution:
University of Bath

Statistical Image Analysis in Cone-beam Computed Tomography

submitted by

Susan Kay Doshi

for the degree of Doctor of Philosophy

of the

University of Bath

Department of Mathematical Sciences

November 2013

ACKNOWLEDGEMENTS

I would like to start by thanking my supervisor, Prof. Chris Jennison. From chatting in a pub about the idea of going back to study to submitting this thesis, Chris has been forever patient—even when I've had my most 'physicist' moments. I would also like to thank Dr Merrilee Hurn, who read all my progress reports and asked a lot of very useful questions at the subsequent meetings.

I have had financial support from the University of Bath, in the form of a University Studentship, which made leaving my job an option.

I've met many wonderful people while I have been working on this. I've had a lot of fun hanging out with the kids, and getting to know the oldies too.

Friends from Hay Hill and NCT have meant a lot. Amongst these, a few stand out. Claire: I have massively appreciated your efforts to keep in touch. Steph and Hazel: you have given me hope that it is possible.

Leela arrived towards the end of this process. From watching the Olympics on 12-hour delay while feeding a tiny scrap of a thing, to watching her career up and down the stairs cackling away, she has kept me thoroughly entertained.

And, last but very definitely not least: Sam. I simply could not have done this without you. You have done so much more than your fair share of everything. Your technical contribution is acknowledged in the text, but more than that I am so very glad that you are interested, that I can bat ideas off you, that you make the effort

to understand, and that you have the sort of brain that asks questions that I don't. And that you care enough about typography to be bothered by this orphan.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Cone-beam computed tomography (CBCT) is used to verify the patient's position prior to commencing radiotherapy treatment. Soft tissues such as the prostate are hard to distinguish, and so gold markers may be implanted. These markers cause artefacts in the 3D reconstruction.

In this thesis, we apply statistical image analysis techniques to CBCT data, with two purposes: we estimate the marker locations (with an assessment of uncertainty), and create reconstructions with fewer artefacts.

In our first analysis, we define a Bayesian statistical model for the projection data, encouraging local smoothness in the prior. We use estimates of the true projection images (generated using Markov chain Monte Carlo, MCMC) in a conventional 3D reconstruction. The results are visually superior to those obtained using a frequency-domain smoothing filter.

In our second analysis, we model the markers as they appear in the projection images. We restrict our model to regions of interest generated using morphological analysis. We combine the information from many projection images to generate an accurate estimate of the marker locations in 3D space. This produces accurate estimates of marker location, but no accurate measure of uncertainty.

Our third analysis uses a template model for the markers in 3D space, with a separate model for the patient tissues. In phantom experiments, we obtain accurate estimates of the tissue properties and marker locations. For practical

computational reasons, we can only analyse a small volume of the patient. Artefacts in the reconstruction used to determine the tissue properties outside the volume of interest prevent the successful estimation of both the tissue properties and marker locations in patients, but we accurately estimate the marker locations alone, with estimates of uncertainty.

Additionally, we process the projection images, removing the markers. These processed images can be used to generate reconstructions with fewer artefacts.

CHAPTER 1

INTRODUCTION

Prostate cancer is the most common cancer affecting men in the UK [6], and the second highest cause of cancer mortality [5]. One widely-used treatment is external beam radiotherapy, where the prostate gland is irradiated using a beam of high-energy (megavoltage, MV) X-rays generated by a linear accelerator.

This thesis will describe the use of statistical image analysis in the context of radiotherapy treatment. The particular imaging modality of interest is computed tomography (CT). Here, the image data are X-ray projections through the patient; in cone-beam CT (CBCT), the projections are 2D images. The information in these projection images may be combined to create a 3D representation of the patient.

Gold markers are implanted in the prostate, allowing the assessment of day-to-day movement of the soft tissues in the pelvis. We shall describe automated methods for finding the position of the markers. The markers cause artefacts in the images. We shall use the statistical analyses to reduce or remove these artefacts.

In Chapter 2, we shall present background information concerning prostate radiotherapy, the clinical problem of locating the prostate gland, and how fiducial markers and cone-beam computed tomography can be used. We shall describe various CT reconstruction methods, and demonstrate some of the problems which arise with these. We shall examine statistical image analysis methods in Chapter

3, describing how they can be applied to recovery of a true scene in both directly observed and tomographic images.

In the subsequent three chapters, we shall show how a number of different Bayesian statistical models can be used to improve the reconstructions, locate the markers, and to remove the artefacts they cause. In Chapter 4, we shall use a Markov random field model for each 2D projection image in turn, and then use conventional methods to carry out the CT reconstruction. The markers will be explicitly modelled in Chapter 5, again in the space of the 2D projections. In Chapter 6, we shall model the markers in the real 3D space of the patient. In Chapter 7, we shall review the thesis.

The sampling in Chapters 4 and 5 was implemented in Python. The 3D model used in Chapter 6 requires a much more computationally intensive approach. I devised the method, and implemented parts of it in Python. It became apparent that the time required for computation would be prohibitive. I then implemented the sampler in C++ in collaboration with Samir Doshi. This implementation ran fast enough to refine the method.

The FDK reconstructions in this thesis have been created using my C++ implementation. However, much of the development of the statistical models and other early work was carried out using MIDAS [29]. This software was developed by colleagues at the INVERT centre at the University of Bath.

CHAPTER 2

CONE-BEAM COMPUTED TOMOGRAPHY IN PROSTATE RADIOTHERAPY

2.1 Radiotherapy for the treatment of prostate cancer

In this chapter, we give some background concerning radiotherapy and cone-beam computed tomography (CBCT) imaging in the context of that treatment. There are several text books on radiotherapy and radiotherapy physics which cover this material, for example [28].

Fig. 2.1 shows two illustrations of a linear accelerator of the type used for radiotherapy. In order to spare normal tissue, the patient is treated with the beam directed from a number of angles, typically four. Tissue along the path of a beam receives a radiation dose, but the highest dose is received where all beams coincide—at the prostate. Each beam is also collimated to the shape of the prostate to minimise the irradiation of normal tissue. For radiobiological reasons, the treatment is most likely to be successful (i.e. gives the highest probability of tumour control, whilst not resulting in unacceptable side-effects) if it is split over a number of days. Patients are typically treated every week day for several weeks.

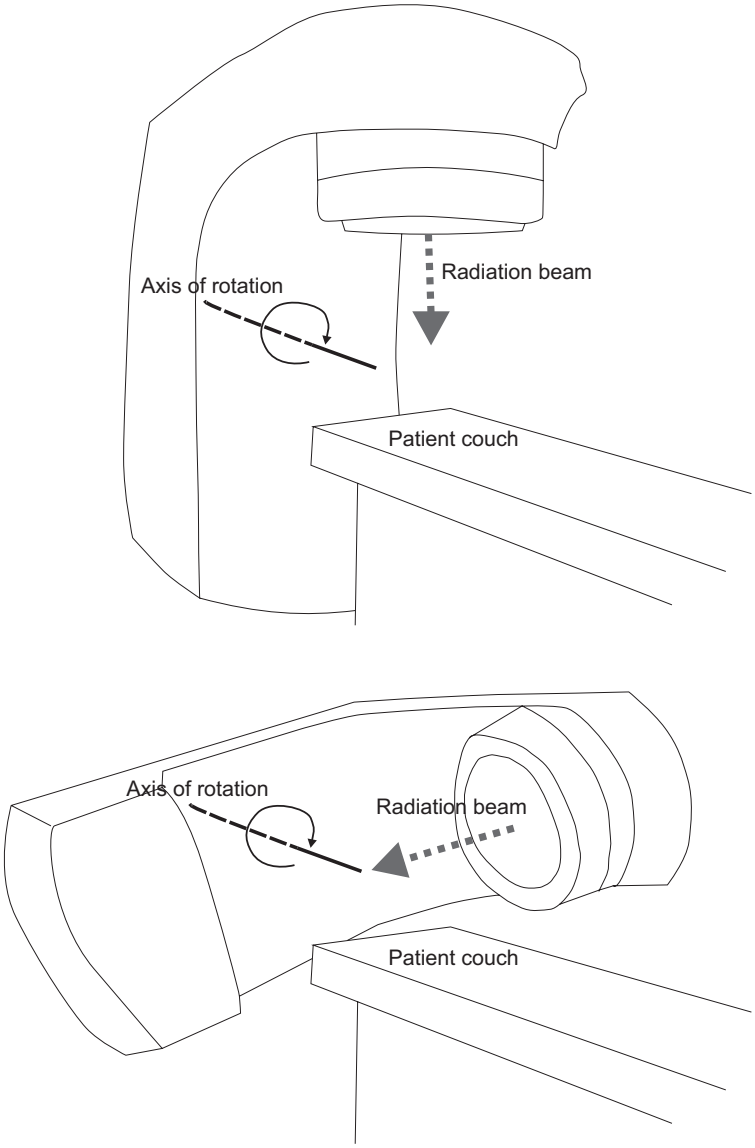


Figure 2.1: Line drawings of a linear accelerator.

Each drawing shows the radiation beam directed towards the patient from a different angle.

Once referred for radiotherapy treatment, the patient undergoes a computed tomography (CT) scan, known as the 'planning' CT scan. The physics and mathematics of CT scanning and reconstruction will be discussed in section 2.2; for now it is sufficient to know that the scan provides a 3D voxellised representation of the patient, with the value at each voxel related to how much X-ray radiation is absorbed or scattered by the tissue in that voxel. This representation of the patient is used to plan the treatment, i.e. to devise a set of treatment beams which will deliver the desired radiation dose evenly to the target volume, whilst delivering as little as possible to the surrounding normal tissue (particularly the bladder and rectum—radiosensitive tissues which are adjacent to the prostate). Parameters that may be adjusted are the angles and relative intensities of the beams, how they are collimated, whether they are 'flat' or 'wedged' (with an intensity gradient from one edge of the beam to the other), and if so how much of a gradient there should be. Producing a treatment plan is a skilled job, and requires significant computational resource—the propagation of the treatment beams through the tissue is modelled on an individual patient basis using the data from the planning CT scan, giving an accurate prediction of the dose distribution in the patient. The process typically takes several hours.

Having spent considerable time and effort to devise an individualised plan, it is important that the patient can be positioned reproducibly in the treatment room. To this end, lasers are mounted on the walls of the room in which the planning CT scan is acquired; small tattoos are applied to the patient at the points where the laser beams intersect the patient's skin (typically three tattoos). By lining up the laser dots with the tattoos, the patient's position can then be reproduced with respect to 'world coordinates' in the same scan room, or, more importantly, in a treatment room equipped with similar lasers.

Tattoos allow us to reproduce the position of the skin surface. Ultimately, we are interested in reproducing the position of the internal organs, particularly the prostate, bladder and rectum. Using lasers and tattoos to position the patient can only ever be as accurate as the displacement of the skin relative to these organs. Once the patient has been set up according to the lasers, the position of the internal anatomy can be verified using some sort of imaging immediately prior to treatment.

There are several modalities used for this 'on-set' imaging, some of which will now be described.

The high-energy (megavoltage, MV) X-ray treatment beam can be used to produce projection images of the patient by measuring the X-ray fluence using a detector on the beam-exit side of the patient. The 3D structure of the patient is projected onto the 2D detector. This is known as portal imaging, and can be thought of as taking a plain projection X-ray image, but using the treatment beam instead of the more usual lower energy X-ray beam produced by a diagnostic X-ray tube. Portal imaging has several advantages. It requires little extra equipment—the treatment beam is already being generated, so all that is required is a detector. It also exposes the patient to little (if any) extra radiation dose, as images can be acquired as the patient is being treated. The first few percent of the radiation treatment may be used to produce an image, with the treatment then interrupted to check positioning. If the position is not satisfactory (i.e. not as it was in the planning CT scan), corrective action may be taken before the rest of the radiation is delivered. An example of a portal image is shown in Fig. 2.2.

If the primary purpose of irradiating the patient were to acquire an image, a lower energy (kilovoltage, kV) X-ray beam would be used instead of the treatment beam. This is because projection imaging using lower energy X-rays results in much more contrast between tissues. The amount by which an X-ray beam is attenuated by an object depends on the length of the path of the X-ray beam through the object, what the object is made of (chemical composition and density), and the energy of the X-ray photons of which the beam is comprised (with a lower energy X-ray beam being scattered and absorbed more than a higher energy one). X-ray projection imaging relies on differences in physical density and in chemical composition to differentiate between tissues. The differences in attenuation caused by differences in chemical composition are greatest for low-energy X-ray beams. For imaging, we want to exploit these differences to give the maximum possible contrast; whereas for therapy we wish to use a beam which is highly penetrating and which gives a reasonably uniform dose distribution regardless of what tissue it has to go through.

In order to obtain images with high contrast between tissues, a kV X-ray tube and a second detector can be mounted on the gantry of the linear accelerator. These

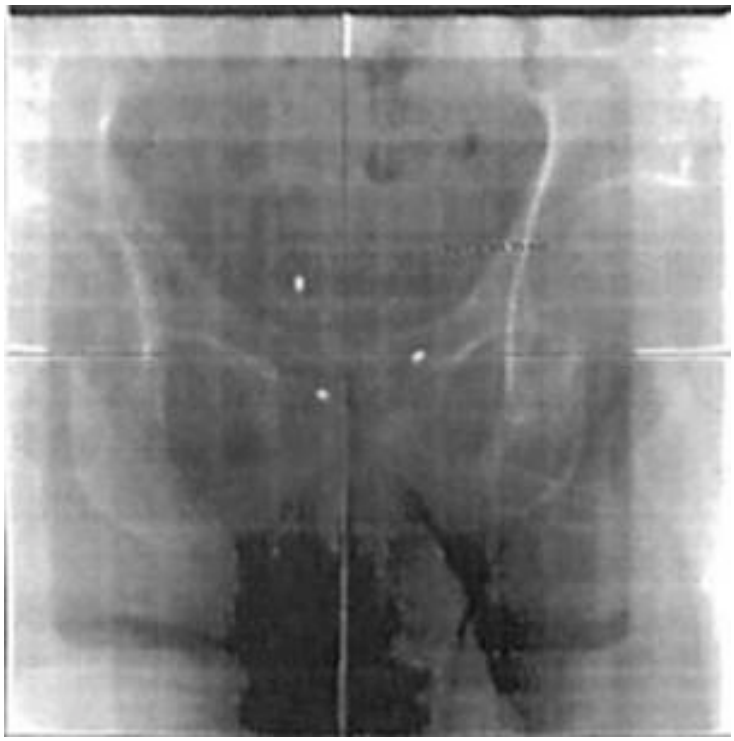


Figure 2.2: A portal image.

This 2D projection image was acquired using a high-energy treatment beam. Three marker seeds (not the Visicoil markers used for most of the data in this thesis) have been implanted. This image was acquired at Bristol Haematology and Oncology Centre.

Figure 2.3: A kV projection image.

A projection image acquired using a kV X-ray tube. Two Visicoil markers have been implanted.

are positioned at 90° to the treatment head (where the treatment beam emerges from the linear accelerator) and portal imaging detector. This adds complexity to the system—keeping the central axes of two X-ray beams within millimetres of the rotation isocentre of the gantry is not a trivial task—but produces images with much higher contrast. Bones are easily visualised, and the X-ray 'shadows' of some large soft-tissue structures can be seen. An example of a kV X-ray projection image is shown in Fig. 2.3. The images in this thesis were acquired at Bristol Haematology and Oncology Centre, using a Synergy (Elekta, Stockholm) kV imaging system.

Using on-set imaging, the position of some internal anatomical structures may be determined relative to world coordinates. Even with kV imaging, the contrast is still not high enough to differentiate between the soft-tissue structures (the organs in which we are actually interested) in the pelvis. Instead, the bony anatomy of the

pelvis may be used as a surrogate. In some other cases, for example the brain, this works very well—the brain does not not move very much with respect to the skull. Unfortunately, the soft tissues within the pelvis are highly mobile. The differences in day-to-day displacement of the bone and prostate (largely due to different filling of the bladder and rectum) can be in excess of 10mm in some cases [4]. If treatment proceeds with these displacements, there may be consequences for the probability of tumour control (if the treatment beam is not covering the intended part of the prostate) and for the severity of side-effects (if there is a high dose of radiation delivered to other radiosensitive organs).

To improve the ability to determine the position of the soft tissues, fiducial markers may be implanted into the prostate itself. In order to be detectable on MV images, these markers must be very radio-opaque. In Bristol, Visicoil (IBA Dosimetry, Schwarzenbruck, Germany) markers are used. These are tightly-wound coils of fine gold wire, two of which are implanted into the prostate. The coils are 0.75mm in diameter; one marker is 10mm long and the other 20mm (the difference in length allowing the markers to be differentiated from each other).

MV portal imaging and kV imaging as described so far in this chapter are 2D projection imaging modalities: the 3D structure of the patient is projected into a 2D image. In each projection image, the pixel value is related to the integral of the linear attenuation coefficient along the path between the X-ray source (linear accelerator or X-ray tube) and the point in the detector which the pixel represents. However, the fact that the linear accelerator and X-ray tube are mounted on a rotating gantry allows tomographic imaging, where projection images from many angles round the patient may be combined to infer the 3D structure of the patient.

2.2 Computed tomography

X-ray computed tomography (CT) imaging involves acquiring information about the transmission of X-ray radiation through an object, and using that to deduce the structure of the object. There are many textbooks discussing the physics, engineering and mathematics of CT, for example Buzug [3] (more applied), Herman [21] and Natterer [31] (more mathematical).

In this section, we shall start by considering conventional (i.e. non-cone-beam) CT. We shall describe how one-dimensional projections of the patient attenuation may be combined to form 'slices'—transaxial representations of the patient through which the projections were acquired. In this imaging modality, the patient is translated through the bore of the scanner in order to acquire different slices. We shall then discuss CBCT, where the detector is a 2D array, and we are interested in reconstructing a 3D representation of the patient.

The CT imaging hardware consists of a gantry which rotates around the patient, on which are mounted an X-ray tube and a detector. In most cases, this is a solid-state detector. The signal at pixel (u, v), $I(u, v)$, is the charge measured in the detector when the patient is irradiated with the gantry positioned at a particular angle. Under certain assumptions, some of which will be discussed later in this chapter, $I(u, v)$ is given by Beer's law. This states that

$$I(u, v) = I_0(u, v) \exp\left\{ - \int_{\mathcal{L}_\theta(u,v)} x \, \mathrm{d}l \right\}, \tag{2.1}$$

where $I_0(u, v)$ is the X-ray signal in the absence of any attenuating patient, and $\mathcal{L}(u, v)$ is the path between the X-ray source and pixel (u, v) in the projection at angle θ.

We typically work with normalised, log-transformed data, which we shall denote P. Thus,

$$P = -\log\left(\frac{I(u, v)}{I_0(u, v)}\right) = \int_{\mathcal{L}_\theta(u,v)} x \, \mathrm{d}l. \tag{2.2}$$

2.2.1 The CT problem

We shall first consider how we can determine the structure of a 2D object from 1D projections through the object. Consider an object represented by a 3×3 grid of square pixels, as shown in Fig. 2.4. Each pixel has a value associated with it, in this case in the range $[0, 100]$. We shall denote these pixel values by x. For now, we shall consider the 'ray sums' in the absence of noise, and in an ideal case with an infinitesimally narrow X-ray beam impinging on an infinitesimally small detector. The ray sums are the sums of the pixel values through which each ray

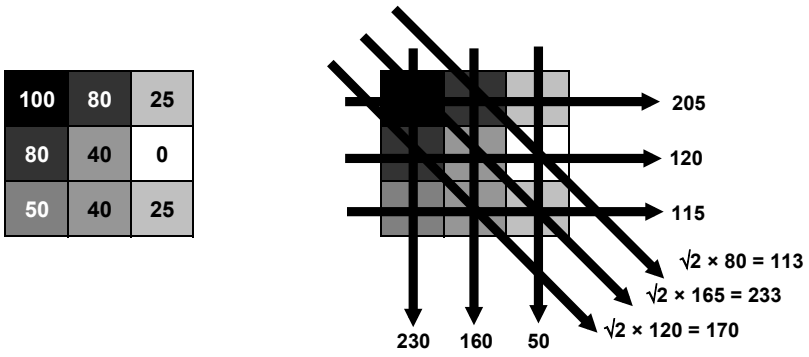


Figure 2.4: An illustrative representation of the CT problem.

The 2D object on the left can be characterised by a collection of ray sums, as shown on the right.

passes, multiplied by the path length of the ray through that pixel, as illustrated in Fig. 2.4. Comparing this figure to Equation 2.2, we can see that the ray sums are a discretised version of P.

We shall call the ray sums y, and represent the system by the linear equation $y = Ax$. The (i, j)th element of A, $a_{i,j}$, represents the path length through the ith pixel in the jth ray.

In this toy example, we have

$$A = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & \sqrt{2} & 0 & 0 & 0 & \sqrt{2} & 0 \\ \sqrt{2} & 0 & 0 & 0 & \sqrt{2} & 0 & 0 & 0 & \sqrt{2} \\ 0 & \sqrt{2} & 0 & 0 & 0 & \sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$x = \begin{pmatrix} 100 \\ 80 \\ 50 \\ 80 \\ 40 \\ 40 \\ 25 \\ 0 \\ 25 \end{pmatrix} \quad \text{and} \quad y = \begin{pmatrix} 205 \\ 120 \\ 115 \\ 113 \\ 233 \\ 170 \\ 50 \\ 160 \\ 230 \end{pmatrix}.$$

Given data y and design matrix A, standard numerical methods may be used to solve this for the pixel values x. However, several problems arise. Firstly, the system is large. For a typical 2D reconstruction, we are interested in 10^5 pixel values, and 10^6 data values. Whilst not completely intractable with modern computational power, the size of the system influences the methods that can be reasonably implemented.

Secondly, the data are subject to noise, and may be inconsistent (that is, there is no x which satisfies $y = Ax$). In the presence of noise, we rewrite the linear system as $\mathbb{E}[Y] = Ax$, where $\mathbb{E}[Y]$ is the expectation of the random variable Y, a realisation of which is the measured data. The solution may be considered to be the value $x^\star$ which minimises $|Ax - y|$, where $|\cdot|$ is the Euclidean norm. Thirdly, the system is very overdetermined, with A ill-conditioned. This means that the solution $x^\star$ is unstable—small differences in the data due to noise may lead to very different solutions—and that some numerical methods may not converge to a solution close to the optimal.

For these reasons, it is usual to implement some sort of regularisation, so that we seek the value x^* to minimise $|Ax - y| + \varphi(x)$ or $|Ax - y|^2 + \varphi(x)$. The penalty function $\varphi(x)$ is chosen to stabilise the system, making the solution less sensitive to the noise in the data. This is achieved by making assumptions about the likely solution. $\varphi(x)$ typically penalises 'roughness' by penalising differences between neighbouring pixels, or the second differences. Some methods penalise the distance from a reference image. Quadratic penalty terms are often used, as these lead to efficient and robust numerical algorithms.

If we assume the the errors in the measurement are independent and normally distributed with a common variance, we can interpret the search for $x^\star$ as finding an estimator based on a penalised likelihood, or a maximum *a posteriori* estimate based on a Bayesian model (for example, the commonly-used method of Tikhonov regularisation can be shown to be equivalent to the statistical method of ridge regression).

Methods of CT reconstruction based on this linear system have been used and investigated since the earliest days of CT. They are often known as 'iterative' methods, because of the way in which they find a sequence of images $x^{(n)}$ which converge to $x^\star$. Different methods are characterised by the way in which $x^{(n+1)}$ is generated from $x^{(n)}$; these are chosen to be computationally efficient (often requiring a convenient subset of the data to be stored in memory at any one iteration), and to minimise the number of iterations required. Iterative reconstruction methods are starting to be used in commercial CT scanners (for example, the Siemens 'IRIS' and GE 'ASIR' algorithms).

2.3 Fourier-based reconstruction

So far, the CT problem has been presented as a linear system, with no particular emphasis on the geometry. In fact, there are standard CT acquisition geometries which make it possible to carry out fast reconstruction using an algorithm called filtered back-projection. This is the work-horse method for most commercial scanners, and will be used as a 'black box' in later chapters. The notation used in this section has been chosen to be consistent with other chapters; however it should be noted that these are not the conventions used by most authors on CT reconstruction.

2.3.1 Parallel beam data scan geometry

The simplest case to consider is that of parallel beams in 2D. This is the geometry that was used in the first CT scanners, in the early 1970s, and is shown in Fig. 2.5. As in the toy example discussed in Section 2.2.1, we seek to reconstruct a 2D object; in reality this is a 2D transaxial slice through a 3D object. If we wish to reconstruct

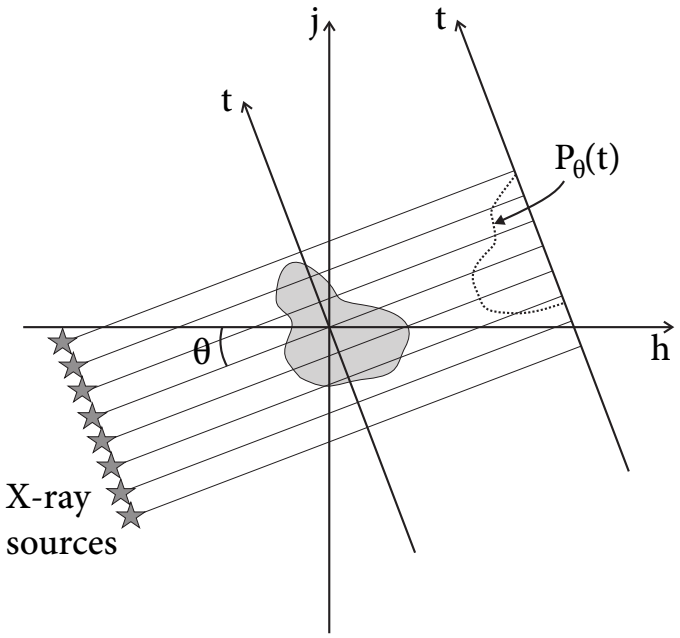


Figure 2.5: The geometry for 2D parallel-beam CT acquisition.

a different slice (to form a 3D representation of the object), we must translate the object though the scanner parallel to the axis of rotation. The 2D object of interest is shown in grey in the centre of the diagram. A set of parallel X-ray beams are incident on this object at an angle θ. The amount by which each of these X-ray beams is attenuated is measured at the appropriate point along a detector, with the distance along the detector indexed by t. We therefore have multiple 1D projections, each of which is acquired at a different angle.

From this set of 1D projections, we wish to create a representation of the 2D object. We shall reconstruct the linear attenuation coefficient, which we denote by $x(h, j)$ at a point (h, j) in a Cartesian coordinate system. The coordinates describing the detector are t (the distance along the detector) and θ (the angle at which the projection is acquired). We consider incident radiation flux I_0, and assume that

Beer's law (Equation 2.1) holds. That is,

$$I = I_0 \exp\left\{ -\int_l x(h,j)\,\mathrm{d}l \right\}$$
$$\implies \int_l x(h,j)\,\mathrm{d}l = -\ln\frac{I}{I_0}. \tag{2.3}$$

We rewrite this to make the link with the projections explicit. We consider the projection data, defined as

$$p_\theta(t) = -\ln\frac{I}{I_0} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x(h,j)\delta(j\cos\theta - h\sin\theta - t)\,\mathrm{d}h\,\mathrm{d}j,$$

where $\delta(s)$ is the Dirac delta function. This is known as the 2D Radon transform of the object. If we consider a single point in the 2D object, the contributions to the projection data in the Cartesian (θ, t) space from this single point form a sinusoidal curve, so this is often called sinogram space.

Let $\mathcal{F}_1[\mathfrak{g}_1](\tau)$ be the 1D Fourier transform of a 1D function $\mathfrak{g}_1(t)$; that is

$$\mathcal{F}_1[\mathfrak{g}_1](\tau) = \int_{-\infty}^{+\infty} \mathfrak{g}_1(t)\exp\left\{ -2\pi\mathrm{i}(t\tau)\right\}\mathrm{d}t.$$

Similarly, let $\mathcal{F}_2[\mathfrak{g}_2](\eta,\iota)$ be the 2D transform of a 2D function $\mathfrak{g}_2(h,j)$; that is,

$$\mathcal{F}_2[\mathfrak{g}_2](\eta,\iota) = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \mathfrak{g}_2(h,j)\exp\left\{ -2\pi\mathrm{i}(h\eta + j\iota)\right\}\mathrm{d}j\,\mathrm{d}h.$$

For CT, we are particularly interested in $\mathcal{F}_1[p_\theta](\tau)$ and $\mathcal{F}_2[x](\eta,\iota)$. The Fourier slice theorem gives a relationship between these: $\mathcal{F}_1[p_\theta](\tau)$ can be identified with a radial line (at angle θ) in $\mathcal{F}_2[x](\eta,\iota)$. That is,

$$\mathcal{F}_2[x](-\tau\sin\theta, \tau\cos\theta) = \mathcal{F}_1[p_\theta](\tau). \tag{2.4}$$

Note that a method of reconstructing $x(h,j)$ from the projections $p_\theta(t)$ is immediately suggested: for a particular (η,ι), find θ and τ such that $\eta = -\tau\sin\theta$ and $\iota = \tau\cos\theta$. The value of $\mathcal{F}_2[x](\eta,\iota)$ is given by $\mathcal{F}_1[p_\theta](\tau)$, which we can find by calculating the transform of the measured projection data. If we do this for each (η,ι), we can then use the 2D inverse Fourier transform to recover $x(h,j)$.

This method of reconstruction is relatively quick, but is not ideally suited to discrete data. This is because $\mathcal{F}[x](\eta,\iota)$ is constructed as radial lines, but a Cartesian grid is required for the discrete 2D inverse transform to recover $x(h,j)$, meaning that regridding is required. Whilst it is possible to measure the projections in such a way that the sample points fall naturally on a Cartesian grid in Fourier space, the method of filtered back-projection is almost always used instead.

To derive the filtered back-projection algorithm, we consider the 2D inverse Fourier transform of $\mathcal{F}_2[x](\eta,\iota)$,

$$x(h,j) = \frac{1}{2}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} \mathcal{F}_2[x](\eta,\iota)\exp\left\{2\pi\mathrm{i}(h\eta + j\iota)\right\}\mathrm{d}\iota\,\mathrm{d}\eta. \tag{2.5}$$

We can change from the Cartesian coordinates (η,ι) to polar coordinates (τ,θ), writing $\eta = -\tau\sin\theta$ and $\iota = \tau\cos\theta$. This implies that $\mathrm{d}\eta\,\mathrm{d}\iota = |\tau|\,\mathrm{d}\tau\,\mathrm{d}\theta$, and the limits of integration over θ become 0 and 2π. From the Fourier slice theorem (Equation 2.4), we know that $\mathcal{F}_1[p_\theta](\tau) = \mathcal{F}_2[x](-\tau\sin\theta,\tau\cos\theta)$. Substituting into 2.5,

$$\begin{aligned}
x(h,j) &= \frac{1}{2}\int_{\theta=0}^{2\pi}\int_{\tau=-\infty}^{+\infty} \mathcal{F}_1[p_\theta](\tau)\exp\left\{2\pi\mathrm{i}\tau(j\cos\theta - h\sin\theta)\right\}|\tau|\,\mathrm{d}\tau\,\mathrm{d}\theta \\
&= \frac{1}{2}\int_{\theta=0}^{2\pi}\int_{\tau=-\infty}^{+\infty}\int_{t=-\infty}^{+\infty} p_\theta(t)\exp\left\{-2\pi\mathrm{i}\tau t\right\}\mathrm{d}t \\
&\qquad\qquad \exp\left\{2\pi\mathrm{i}\tau(j\cos\theta - h\sin\theta)\right\}|\tau|\,\mathrm{d}\tau\,\mathrm{d}\theta \\
&= \frac{1}{2}\int_{\theta=0}^{2\pi}\int_{t=-\infty}^{+\infty} p_\theta(t)\int_{\tau=-\infty}^{+\infty} |\tau|\exp\left\{2\pi\mathrm{i}\tau(j\cos\theta - h\sin\theta - t)\right\}\mathrm{d}\tau\,\mathrm{d}t\,\mathrm{d}\theta
\end{aligned} \tag{2.6}$$

We now write

$$g_\infty(t) = \frac{1}{2}\int_{-\infty}^{+\infty} |\tau| \exp\left\{\pi i \tau t\right\} \mathrm{d}\tau. \tag{2.7}$$

This defines a ramp filter $g_\infty(t)$. This is a high-pass filter—that is, it attenuates low spatial frequencies (removing the zero-frequency component entirely). It is known as a 'ramp' filter because of its shape in Fourier space: $\mathcal{F}_1[g_\infty](\tau) \propto |\tau|$.

Let the convolution operator $\otimes$ be defined such that

$$(f \otimes g)(\tau) = \int_{-\infty}^{\infty} f(t) g(\tau - t)\, \mathrm{d}t. \tag{2.8}$$

Using this operator and substituting the definition in Equation 2.7 for $g_\infty(t)$ into Equation 2.6, we obtain

$$x(h,j) = \int_{\theta=0}^{2\pi} (p_\theta \otimes g_\infty)(j\cos\theta - h\sin\theta)\, \mathrm{d}\theta. \tag{2.9}$$

This now suggests another algorithm for reconstruction of $x(h,j)$: acquire the projection data $p_\theta(t)$ and apply the ramp filter $g_\infty(t)$. Now perform back-projection: for each point in the filtered image, 'smear' (or back-project) the value at this point along the line back to the X-ray source (the line $h = -\tau\sin\theta, j = \tau\cos\theta$), adding the value of the filtered image to the value in the reconstruction at each point along the line. Do this at every angle.

(We have described the application of the ramp filter as a convolution in real space; in practice, it is in fact applied in Fourier space. This is an application of the convolution theorem, which states that the convolution of two functions is equal to the inverse transform of the product of the transforms of the functions.)

In the continuous case, this reconstruction is identical to the direct Fourier method described above up to an additive constant (which is lost in the application of the ramp filter). The advantages of this algorithm with real data are firstly that there is no regridding required, and secondly that the filtering can be carried out

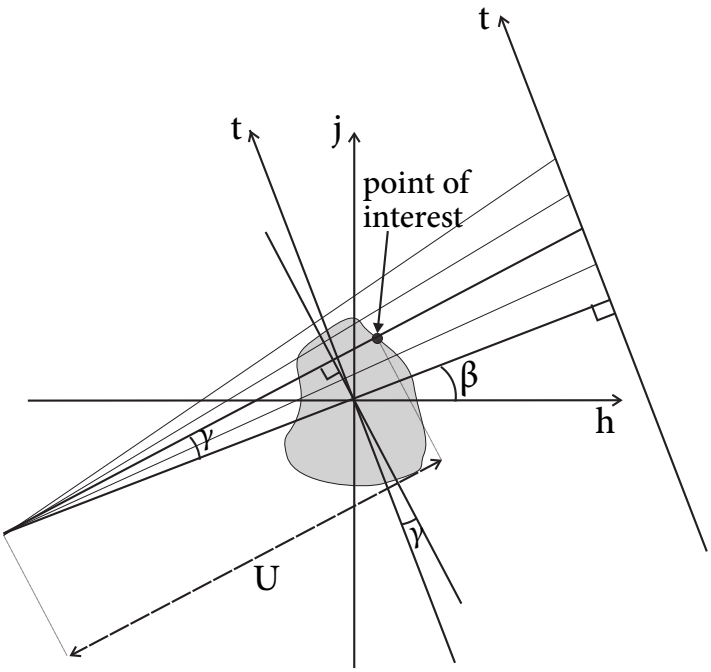


Figure 2.6: The geometry for 2D fan-beam CT acquisition.

on each projection as soon as it is acquired—there is no need to wait until all projections are acquired before the 'heavy lifting' can start.

This algorithm is is known as *filtered back-projection* and it is the basis of the reconstructions used by nearly all commercial CT scanners. In practice, the high-pass ramp filter enhances noise. Low-pass filters are used in addition to the ramp filter to decrease the contribution of high spatial frequencies to the final reconstruction.

2.3.2 Fan-beam scan geometry

Parallel beam geometry requires either multiple X-ray sources, or for a single X-ray source to be translated along a linear path before each rotational movement. This is very time consuming, and was superseded fairly quickly in medical CT (although it is used in other applications, such as micro-CT of specimens where the object, rather than the X-ray source, is rotated). Instead, a fan beam is used, with the

geometry shown in Fig. 2.6. This data-set can be re-ordered into the same as that which would have been acquired with parallel beams relatively easily. In both cases, each point of interest in the object is intersected by a number of beams. In the parallel beam case, these beams were indexed by the angle of the X-ray source, θ. In the fan beam case, we also use the *fan angle*, γ, defined as the angle between the line joining the X-ray source and the centre of rotation and that joining the source and the point of interest. The projection at point t measured at source angle β and fan angle γ is measured along the same line as for a parallel beam at angle $\theta = \beta + \gamma$.

Some systems (in fact most dedicated diagnostic CT scanners) use a curved detector array, so that the source-detector distance remains constant for all values of γ. However, we shall consider a linear detector, as it leads more naturally to the cone-beam geometry. A correction must be made for the differing path lengths through the object. Equation 2.9 is modified to become

$$x(h,j) = \int_0^{2\pi} \frac{R^2}{[U(h,j,\beta)]^2} \tilde{p_\beta}(t)\, \mathrm{d}\beta \tag{2.10}$$

where

$$\tilde{p_\beta}(t) = \left(p_\beta(t) \frac{R}{\sqrt{R^2 + t^2}} \right) \otimes g_\infty(t) \tag{2.11}$$

for projection data $p_\beta(t)$, source-detector distance R, $g_\infty(t)$ a ramp filter as in equation 2.7, $U = h\cos\beta + j\sin\beta$, and $t = j\cos(\beta+\gamma) - h\sin(\beta+\gamma)$.

2.3.3 Cone beam CT

Cone-beam CT may be considered as an extension of the fan beam geometry, adding another dimension. Instead of using a fan-beam to irradiate a transaxial slice of the object, a 3D volume is irradiated. Instead of measuring 1D projection data at each angle, a 2D detector array is used to capture the signal. We need to consider the *cone angle* δ as well as the fan angle γ and source angle β. The geometry is shown in Fig. 2.7.

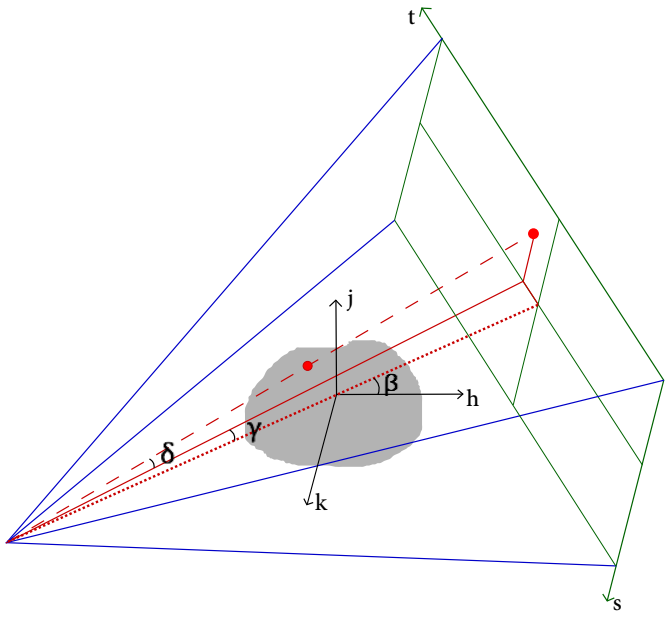


Figure 2.7: The geometry for 3D cone-beam CT acquisition.

The 3D coordinates of real space are labelled h, j and k. The boundary of the detector and the (s, t) detector coordinate system are shown in green. The 'cone beam' (actually a pyramid with the X-ray source at the apex and the detector as its base) is shown in blue. The central axis of the X-ray beam is shown using a dotted red line. The point of interest and the point in the detector at which this point is projected are both shown as red dots. The dashed red line shows the X-ray path through the point of interest.

The source angle β is the angle between the h axis and the central axis of the X-ray beam. The fan angle γ and the cone angle δ are the angles between the dashed red line and the central axis in the planes perpedicular to s and t respectively.

For small cone angles (such as those used on CT scanners with four rows of detectors), the cone angle may be ignored and a good reconstruction created under fan-beam assumptions using Equation 2.10. For larger angles, such as those used by the Synergy system from which the data used in this thesis are obtained, the cone angle must be considered explicitly.

The 2D methods described above are exact (that is, as long as our assumption that Beer's law holds is correct, the object can be reconstructed to an arbitrary resolution by acquiring data at sufficiently high frequency in θ and t). This is a consequence of the Fourier slice theorem: it is possible to fill the Fourier transform of the object at arbitrary resolution by collecting sufficient projection data. In 3D, we would need to fill a 3D Fourier space using 2D projections, and the Tuy-Smith sufficiency condition [36, 38] tells us what projections we need in order to do that. This condition states that an exact reconstruction can be obtained if all planes intersecting the object also intersect the source trajectory, and this is in fact a necessary (as well as sufficient) condition for an exact reconstruction. The circular trajectory used by Synergy violates this condition, and so it is only possible to implement an approximate reconstruction. The most commonly used algorithms are based on those described by Feldkamp, Davis and Kress [13]. These are known as 'Feldkamp' or 'FDK' algorithms, and use a correction to account for the cone angle. As for parallel and fan-bean reconstruction, the algorithm proceeds by applying a ramp (and possibly low-pass) filter to corrected projection data, then integrating along a trajectory in a 3D Radon (sinogram) space.

Let (t, s) be the coordinates in the detector (with t as before and s parallel to the axis of rotation). The back-projection (equation 2.10) can be shown to become

$$x(h, j) = \int_0^{2\pi} \frac{R^2}{U^2} \tilde{p_\beta}(t, s)\, \mathrm{d}\beta \tag{2.12}$$

where

$$\tilde{p_\beta}(t, s) = \left(p_\beta(t, s) \frac{R}{\sqrt{R^2 + t^2 + s^2}} \right) \otimes g_\infty(s, t), \tag{2.13}$$

$t = j\cos(\beta+\gamma) - h\sin(\beta+\gamma)$ and $s = R\tan\delta$ for source-detector distance R.

2.4 CT artefacts

An artefact is any feature of the image which does not represent a feature in the patient. CT artefacts are many and varied, but two types of artefact will be of particular interest in this thesis, as they both occur with fiducial markers. We shall use examples of 2D CT, but the principles explained in this section all apply to CBCT.

2.4.1 Star artefacts

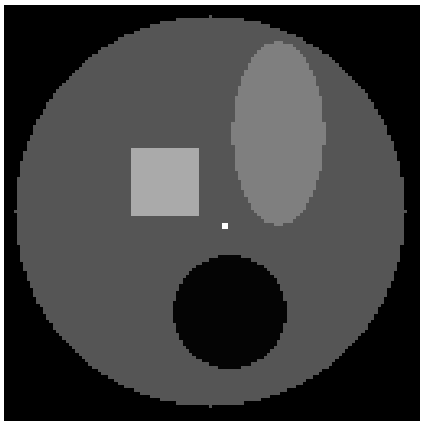
Star artefacts are caused by under-sampling in the angular coordinate, and then applying a ramp filter. The sharp edges of the markers are projected back and cause a star artefact. This is illustrated in a mathematical phantom in Fig. 2.8, and with a patient in Fig. 2.9.

2.4.2 Streak artefacts

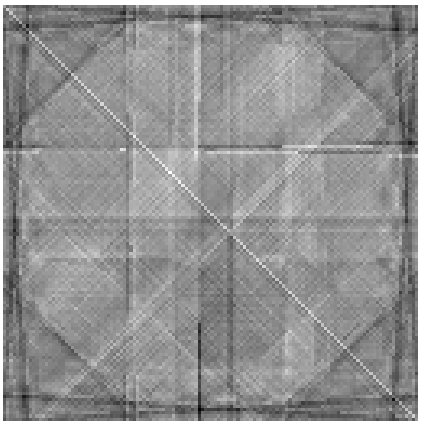
Streak artefacts arise from a more fundamental physical cause. In our discussion of CT reconstruction and filtered back-projection, we have assumed that X-ray attenuation can be described by Beer's law

$$I = I_0 \exp\left\{ -\int_l x(h,j)\,\mathrm{d}l \right\}.$$

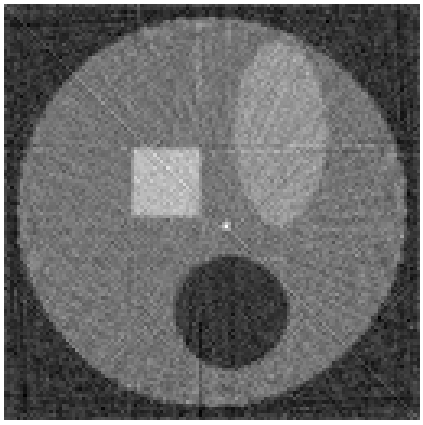
In fact, the linear attenuation coefficient x depends not only on the spatial location (h,j), but also on the energy of the X-ray photons. When we describe an X-ray beam as being of a certain energy (e.g. 80kV, 120kV, 6MV), we are referring to the X-ray tube voltage: the voltage across which a stream of electrons is accelerated before impinging on a tungsten target. It is the interaction of these electrons with the tungsten which produced X-ray photons. An electron accelerated across a voltage $\mathcal{V}$V has an energy of $\mathcal{V}$eV, which is consequently the maximum possible energy with which an X-ray photon can be produced. X-ray photons with energies



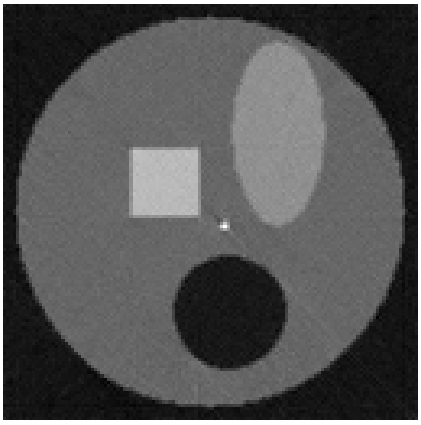
a: A 2D object to be reconstructed from projections.



b: Reconstruction from eight projections



c: Reconstruction from 64 projections



d: Reconstruction from 512 projections

Figure 2.8: Star artefacts in a phantom.

In the reconstructions from eight and 64 projections, lines appear to emanate from the bright object in the centre of the image. These are known as star artefacts.

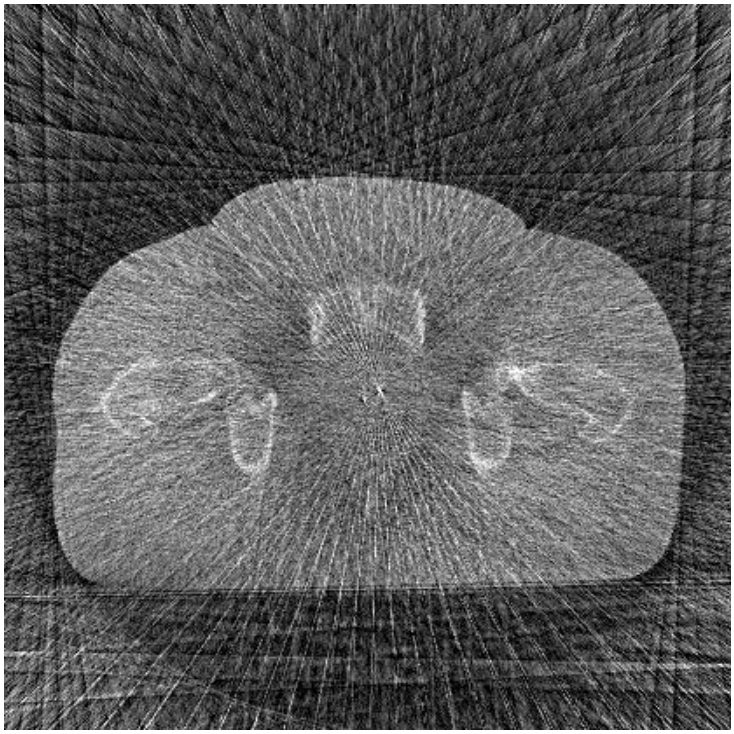


Figure 2.9: Star artefacts in a patient.

The image shows a transaxial slice from an FDK reconstruction of a patient with implanted markers, using a subset (one eighth) of the available data to form the reconstruction. The markers cause star artefacts.

over the whole spectrum from zero up to $\mathcal{V}$eV are present in the X-ray beam, in the same way that white light consists of photons of varying energies. We describe such a beam as polyenergetic.

We now write the linear attenuation coefficient of the material at spatial position (h, j) at energy $\mathcal{E}$ as $x^{(\mathcal{E})}(h, j)$. Let the incident and signal intensities at energy $\mathcal{E}$ be $I_0^{(\mathcal{E})}$ and $I^{(\mathcal{E})}$ respectively. When we consider the polyenergetic nature of the X-ray beam and the dependence of x on the photon energy, Equation 2.1 becomes

$$\int_{\mathcal{E}} I^{(\mathcal{E})} \, \mathrm{d}\mathcal{E} = \int_{\mathcal{E}} \left[I_0^{(\mathcal{E})} \exp\left\{ - \int_l x^{(\mathcal{E})}(h, j) \, \mathrm{d}l \right\} \right] \mathrm{d}\mathcal{E}. \tag{2.14}$$

The signal detected is the integral in Equation 2.14, which can no longer be linearised.

The linear approximation works where the variation in $x^{(\mathcal{E})}(h, j)$ over the range of energies in the X-ray beam are small. This is the case when either the X-ray beam is close to monoenergetic, or the materials of which the patient is comprised are similar in terms of their attenuation properties. The former condition can in some cases be met by filtering out all but a narrow range of photon energies, but this is not practical for general medical systems because the electrical power needed to generate the 'waste' photons which are filtered out would be too great. Instead, a more limited amount of filtration is used which ameliorates the problem for most situations, removing the lowest energy photons. In general, the human body usually consists of soft tissues, bone and air. With a filtered beam, this is a narrow enough range of materials to avoid serious streak artefacts. However, metallic objects are implanted into patients for a number of reasons—apart from the fiducial markers discussed in this thesis, other metallic objects include prostheses, surgical clips and dental fillings. The attenuation properties of these metallic objects are sufficiently different from biological tissue that the linear approximation is no longer a good one.

We shall demonstrate the consequences of applying the linear approximation in the situation where metal is present using a small mathematical 2D phantom, shown in Fig. 2.10. The phantom is 16×16 pixels in size, with each pixel being either air, water, or a mixture of 30% gold and 70%water (by volume). The attenuation

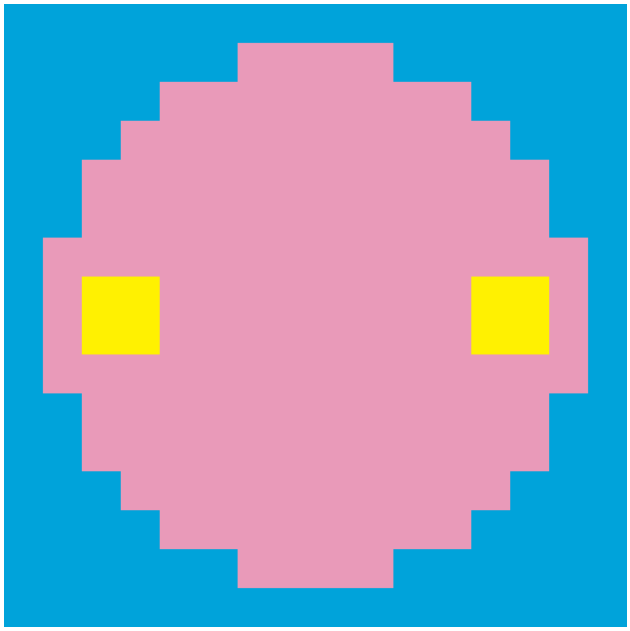


Figure 2.10: 2D object used to illustrate streak artefacts.

The object is 16 × 16 pixels, and consists of air (blue), water (pink), and a mixture of water and gold (yellow).

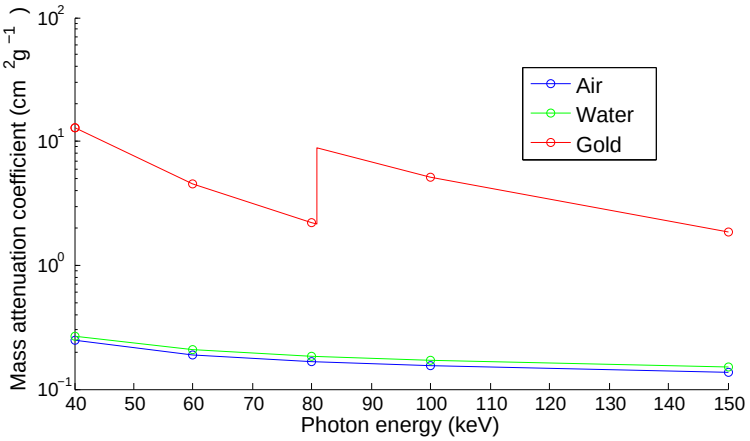


Figure 2.11: Mass attenuation coefficients.

The coefficients of three materials in the range 40keV - 150keV are shown [22]. The mass attenuation coefficient is the linear attenuation coefficient divided by the density of the material, and is used here to enable the attenuation of the three materials to be plotted on one chart.

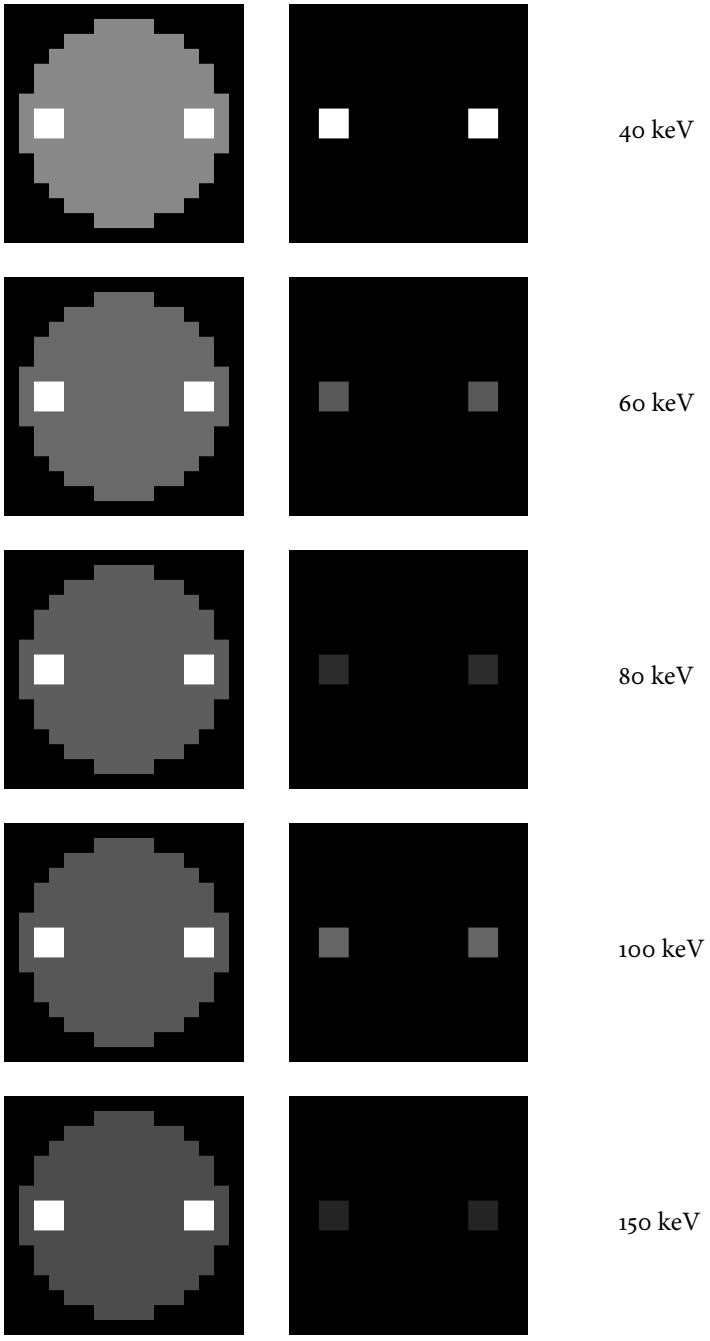


Figure 2.12: The linear attenuation coefficients of the object in Fig. 2.10.

The linear attenuation coefficients of the materials are shown at five energies as indicated for each row. The two columns of images show the same data with different greyscales. In the first column the greyscale has the white level set below the attenuation value for the gold/water mixture; on the right the white level is set to be the maximum for this material (i.e. the value at 40keV).

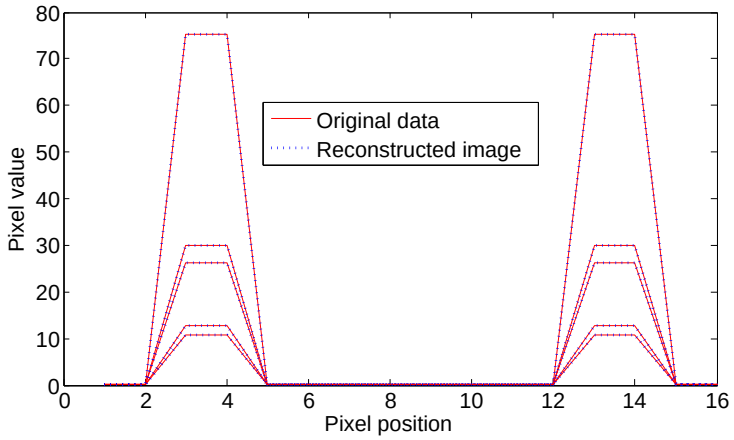


Figure 2.13: Pixel values from simulated monoenergetic projections.

properties of these materials are shown in Fig. 2.11. (The jump in the attenuation of gold at 80.7keV is due to photoelectric absorption being more probable around the K-edge at this energy [39]). The attenuation is clearly not constant with changing energy, and neither is the relationship between the attenuation of the three materials.

Fig.2.12 shows the linear attenuation of this object at five different keV values. 1D projections through the phantom were simulated at 128 angles for each of the five sets of attenuation values. In the language of the example in Section 2.2.1, we have generated the projection data $y = Ax$. Five reconstructions were created by finding the values of the linear attenuation $x^\star$ which minimises $|Ax-y|$. For this example, A is of full rank, so this is simply inverting the process by which the projections were created. The resulting reconstructions are perfect (within numerical precision). This is illustrated in Fig. 2.13, which shows a profile of the pixel values in the original and reconstructed images, for each of the five photon energies.

Polyenergetic projection images were simulated as

$$y_{\text{poly}} = -\log\left[\sum_{\mathcal{E}} \exp\{-y_{\mathcal{E}}\}\right] = -\log\left[\sum_{\mathcal{E}} \exp\{-Ax_{\mathcal{E}}\}\right],$$

where $x_{\mathcal{E}}$ are the linear attenuation values and $y_{\mathcal{E}}$ are the projection values, both obtained at energy $\mathcal{E}$. We use these polyenergetic images to create a reconstruction

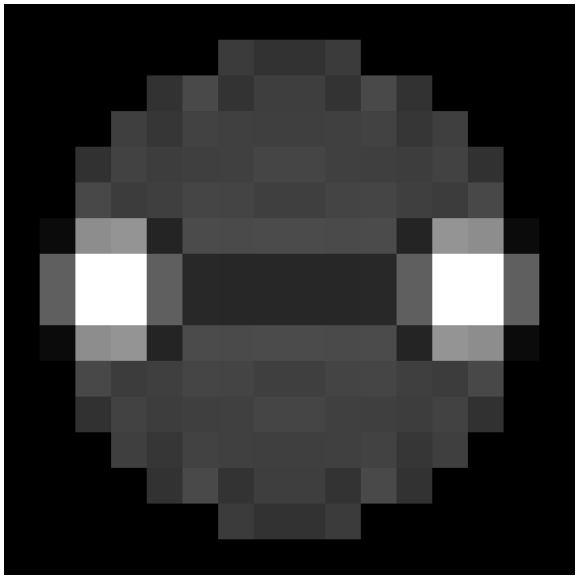


Figure 2.14: Streak artefact in a phantom.

In this reconstruction from simulated polyenergetic projections, there is a dark area between the two bright squares. This is a streak artefact.

in the same way as before—by finding

$$x^\star = \arg\min_x |Ax - y_{\text{poly}}|. \tag{2.15}$$

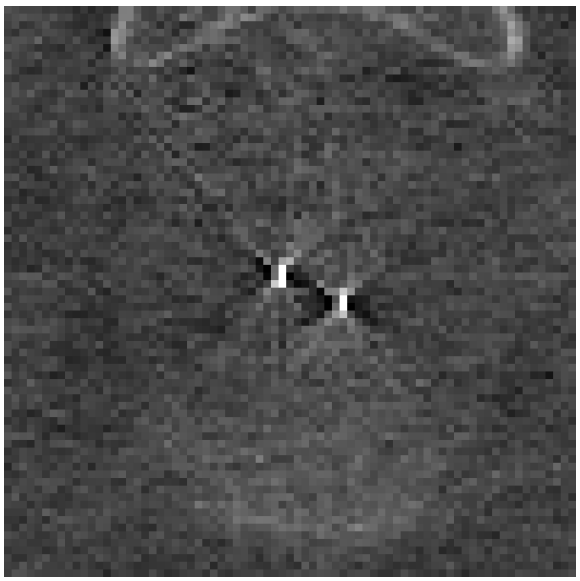


Figure 2.15: Streak artefacts in a patient.

Part of a transaxial slice through an FDK reconstruction of a patient. There is a dark area between the two bright markers. This is a streak artefact.

Fig. 2.14 shows this reconstruction. We are creating a reconstruction by solving a linear approximation to the system. The artefacts in this image are not due to discretisation, filtering or any other aspect of filtered back-projection—they are purely a result of using a linear model to attempt to solve a nonlinear problem. Note particularly the dark area between the two gold/water regions. This is a streak artefact, and in clinical images occurs frequently between areas of high attenuation. Fig. 2.15 shows such an artefact in an FDK reconstruction of a patient with markers implanted.

2.5 Description of data

The data used in this thesis are scans of four patients undergoing treatment at Bristol Haematology and Oncology Centre, three of whom have the markers implanted. My use of the anonymised patient data was approved by the Caldicott Guardian. We shall describe these data sets as 'no markers', 'straight markers', 'bendy markers' and 'hooked markers' (for reasons which will become obvious). Additional scans of blocks of Perspex were also acquired. Perspex has very similar X-ray attenuation properties to soft tissue, and is therefore often used as a general-purpose phantom for X-ray experiments. Different thicknesses of Perspex were placed on the treatment couch and scanned using the usual patient scan settings.

For each patient, the data consist of approximately 600 projection images, acquired at approximately evenly spaced angles over a 360° rotation. The images are 512×512 pixels in size, with pixel pitch 0.8mm. They were acquired using an offset detector, meaning that the central axis of the X-ray beam is not incident at the centre of the detector. This means that the patient appears off-centre (on the right-hand side) in each of the images. For a given size of detector, source-isocentre distance and source-detector distances, using an offset detector allows us to acquire images of a larger patient. The central part of the patient (which we are most interested in) is included in all of the images, whereas the periphery is only included in half of the images, 'cropped' off the right-hand side of each projection image.

The offsets are provided for each projection. These values include the deliberate offset, as described above, but also the 'gantry sag'. The gantry, X-ray tube and

detector are heavy, and the rotation around the isocentre is not perfect. At approximately three monthly intervals, images of a ball bearing positioned at the isocentre are acquired, allowing quantification of the sag. These measurements are included in the offsets for each image, and account for up to a few millimetres. They are, however, only as accurate as the reproducibility of the sag over a three-month period allows.

The statistical and physical models used here are approximations to the truth. The signal measured in pixel (u, v) is related to the number of X-ray photons absorbed in the detector at that location. A series of physical interactions in the detector creates a measurable charge, the magnitude of which depends on the number of X-ray photons absorbed and on their energy. The detector acts as an energy integrator, with the result that the measured signal has a compound Poisson distribution [40]. The signal is amplified, digitised, and has some low-level processing applied (for example, gain and dead pixel corrections). The signal is then quantised, scaled so that it lies in the range $[0, 2^{16} - 1]$, log-transformed, and rescaled. In both rescaling steps, the scaling factor is negative. In this thesis, the phrase 'pixel value' will be used to refer to the result of these transformations, as this is the format in which the data are retrieved.

Rather than attempting to model the detector sufficiently accurately to use the correct compound Poisson distribution, we shall use normal approximations. The scans of Perspex blocks of known thickness demonstrated that the pixel value is approximately linear in the thickness of Perspex. Deviations from linearity may be explained by physical processes not included in our model. These include beam hardening and scattering of X-rays in the scanned object and in the detector.

The variance of the pixel values did show a dependence on the mean, as would be expected for a Poisson or compound Poisson process. However, due to the ill-defined transformations which have been applied to the data and the incompleteness of the physical model, the relationship between the mean and variance of the pixel values does not appear straightforward. The normal models in this thesis use the approximation of a constant variance.

These statistical and physical approximations are not a fundamental requirement of the analyses presented here (we do not rely on convexity for numerical

minimisation, for example). If greater accuracy were required, more realistic models could be used.

2.6 Statement of problem

In this thesis we shall address two problems associated with the use of fiducial markers in CBCT. Firstly, the purpose of implanting the markers is to determine the position of the prostate. We shall assume that the markers do not move relative to the tissue of interest, and so we wish to determine the position of the markers.

If there is no gross error in the position, but it is found to be outside a pre-determined tolerance, the patient couch may be translated and rotated to correct these errors. The required translation and/or rotation can conveniently be calculated by identifying a set of points to be matched. For this reason, it is particularly the location of the ends of the markers that are of interest. In this thesis, our first task will be to generate estimates of the location of each of the marker ends. By using statistical methods, we shall also be able to provide credible intervals for these estimates.

Secondly, the markers cause artefacts as illustrated in Figures 2.9 and 2.15, which can obscure other anatomy. As a result of knowing where the markers are, we shall be in a position to remove these artefacts.

In this chapter, we have given an introduction to CBCT imaging, how it is used in prostate radiotherapy, and outlined the clinical problem. In the next chapter we shall introduce the statistical methods that will be employed to provide some solutions.

CHAPTER 3

STATISTICAL IMAGE ANALYSIS

3.1 Introduction and notation

In this chapter, we shall present some of the statistical models which may be applied to image data. We shall start with a general description, and then detail how some of these have been applied to CT and other tomographic data.

We shall consider an image to be a 2D or 3D representation of some object on a grid of square pixels or cuboid voxels. For the sake of clarity of notation, we shall use the 2D case for our examples. Let the size of the pixel grid be $N_u \times N_v$, so that there are $N_p = N_u \times N_v$ pixels in the image.

We are interested in the 'true scene', $X = \{X_{(u,v)}\}$, where $X_{(u,v)}$ represents the truth about the object at the location represented by pixel (u, v). To give two examples, in digital photography X represents the light reflection, transmission and emission characteristics of the objects in the photograph; in X-ray CT it is the X-ray attenuation properties of the patient. We form an image by using some physical process to make a measurement which gives us information about the object; in both photography and X-ray CT, we observe the signal generated by photons interacting in a detector. We denote this observed signal (i.e. the data) by $Y = (Y_1, \ldots, Y_{N_y})$. In image analysis, we wish to infer information about X from our observations Y.

Many images are 'directly observed', meaning that each element of the data is particularly associated with one pixel in the true scene. For example, a photograph might be corrupted by noise or motion blur, but there is a sense in which the data recorded in one pixel of the detector 'belongs' at that location—there is a one-to-one mapping between X and Y. This is not the case in CT: the true scene X is a 2D (or 3D) representation of the patient, with the data Y being a set of 1D (or 2D) projections through the patient.

We denote the data generation process by Ψ, so that $Y = \Psi(X)$. The function Ψ may describe several aspects of the physical measurement. For example, it may describe blurring processes, such as motion blur or the spread of the signal in the detector, and generally includes a stochastic element.

3.2 Bayesian statistical modelling

In this thesis, we shall adopt a Bayesian statistical approach, in which X and Y are interpreted as random variables. For the analyses presented in later chapters, we shall require a prior distribution for X and the likelihood of Y given X. These define a posterior distribution (the distribution of X given Y), which is used to perform inference on X. We shall now discuss each of these distributions, describing their role in image analysis.

3.2.1 The prior probability of X

The prior probability distribution for the true scene, $\pi(X)$, incorporates information about what we expect to see. This prior information could be large in scale (for example, we might know that we are dealing with an image of a particular anatomical region). One possibility would be to assume, *a priori*, that the true scene is similar to some reference scene z, with $X_{(u,v)}$ having mean $z_{(u,v)}$.

Template models

Another approach to large-scale knowledge is to use a deformable template, as in Grenander *et al.*'s 'Hands' [17]; these models are also described in [9]. Here, we use a parameterisation of a feature we wish to encode, and then specify

the prior probabilities of particular realisations of the feature in terms of this parameterisation. As an illustrative example, we might expect the true scene to be an image of an ellipse. We could parameterise this ellipse by its centre (u_c, v_c), major and minor radii (r_u, r_v), and the angle of the major axis from horizontal θ. We might know approximately what size ellipse to expect, where in the image it was likely to appear, whether it was likely to be of high or low eccentricity, etc. We could then define a probability distribution $\pi(u_c, v_c, r_u, r_v, \theta)$ to encode our prior knowledge. The template approach to image analysis can be extended to much more complicated structures than an ellipse, and will be used in later chapters.

Markov random field models

Instead of large scale features, we may wish to use the prior density to encode information about the local characteristics of the image. For example, we may expect neighbouring pixels in the true scene to have similar values. One way of defining a distribution on X is by using a Markov random field (MRF) model and defining the conditional distribution of each pixel given its neighbours. The use of this approach in image analysis was pioneered by Geman and Geman [14] and Besag [2].

To define an MRF, we first define a neighbourhood for each pixel. In a 2D image, this might be the four pixels sharing a boundary (a first order neighbourhood), or could also include the additional four pixels which touch at the corners (a second order neighbourhood). These are shown in Fig. 3.1. We write $i \sim j$ if the pixels $i = (u_i, v_i)$ and $j = (u_j, v_j)$ are neighbours, and use $\delta(i)$ to denote the pixels in the neighbourhood of pixel i, with $x_{\delta(i)}$ being the values of those pixels. The random field defined by the pixel values is Markovian if the probability of one pixel taking a particular value is conditionally independent of the other pixels, given its neighbours.

We shall use the convention that π will be used to denote the prior distribution, or the conditional probabilities for this distribution (with the names of the arguments of π defining the distribution that is represented). We write $x_{\setminus i}$ is to denote the values of all the pixels except i. We can then write the Markov property as $\pi(x_i|x_{\setminus i}) = \pi(x_i|x_{\delta(i)})$.

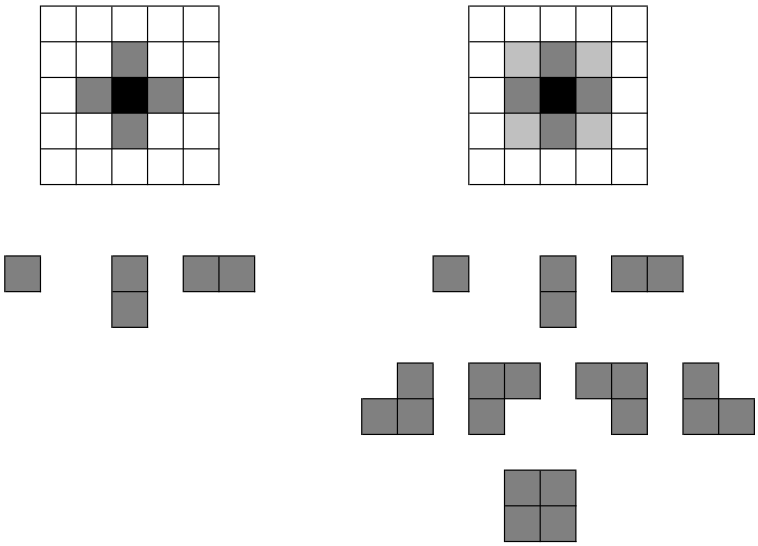


Figure 3.1: Illustration of neighbourhood structures.

Two possible neighbourhood structures for a 2D image are shown. In each diagram, we are considering the neighbourhood of the pixel shaded black, with each of the pixels in the neighbourhood being shaded grey. The first order neighbourhood (left) contains the four pixels sharing boundaries; the second order neighbourhood (right) includes the first order neighbours (dark grey), and also the four pixels touching at the corner (light grey). Shown underneath the neighbourhoods are the associated cliques, in which every pixel is a neighbour of every other pixel.

We require a prior distribution for the true scene, $\pi(x) = \pi(x_1, x_2, \ldots, x_n)$. In order to define this, we shall first define a clique. We can consider the array of pixels in the image to be an undirected graph. Each pixel forms a vertex, with an edge (in the sense of a graph rather than an image feature) existing between two pixels i and j if they are neighbours, i.e. if $i \sim j$. A clique $\mathfrak{c}$ is a subset of pixel indices in which either every pixel is a neighbour of every other pixel, or the subset consists of a single pixel. For a first order neighbourhood in a 2D image, the cliques are either indices representing single pixels, or two pixels sharing a boundary. In a second order neighbourhood, they may also consist of three or four pixels (as shown in Fig. 3.1). Let the set of such cliques be $\mathfrak{C}$.

Suppose our joint distribution for the true scene is of the form

$$\pi(x) = \exp\left\{ -\sum_{\mathfrak{c}\in\mathfrak{C}} \Phi_{\mathfrak{c}}(x) \right\} \tag{3.1}$$

where $\Phi_{\mathfrak{c}}$ depends only on $\{x_i \text{ s.t. } i \in \mathfrak{c}\}$. In this case, $\pi(x)$ is an MRF, with neighbourhoods

$$\delta(i) = \bigcup_{\mathfrak{c}\in\mathfrak{C} \text{ s.t. } i\in\mathfrak{c}} \mathfrak{c}.$$

The conditional probabilities are given by

$$\pi(x_i|x_{\setminus i}) \propto \exp\left\{ -\sum_{\mathfrak{c}\subset\delta(i)} \Phi_{\mathfrak{c}}(x) \right\}. \tag{3.2}$$

In fact, the Hammersley-Clifford Theorem [1] tells us that for any MRF with this set of cliques $\mathfrak{C}$, the joint distribution is of the form given in Equation 3.1. The functions $\Phi_{\mathfrak{c}}$ are called the 'clique potentials' (due to an analogy with statistical physics).

As previously mentioned, the function of the prior distribution is often to encourage smoothness (achieving the same end as the regularisation φ discussed in the last chapter). This is achieved through the action of the clique potentials—realisations with high values for the potentials have a low prior probability. Some suitable choices are $\Phi(x_i, x_j) = |x_i - x_j|$ (i.e. using the $\ell^{(1)}$ norm) or $\Phi(x_i, x_j) = (x_i - x_j)^2$ ($\ell^{(2)}$ norm). The former is better at preserving edges in the image whilst the latter may create a posterior density which is more amenable to maximisation without resorting to sampling-based methods.

3.2.2 The likelihood of Y given X

The likelihood of the data Y given the true scene X incorporates the data generation process Ψ into the Bayesian statistical framework. As an illustrative example, a

possible model for data in a directly observed image might be

$$Y_{(u,v)}|X = x \sim \mathcal{N}(x_{(u,v)}, \sigma^2),$$

where $Y_{(u,v)}$ is the random variable whose realisation is the observed data point $y_{(u,v)}$, and $x_{(u,v)}$ is the corresponding pixel value in the true scene. Thus the observed pixel values are independent given the true scene.

This model describes the image noise; it is also possible to include data generation processes which blur the image—such as motion blur or a sensor with point spread function of finite width—whilst retaining the property that the observed values are independent given the true scene. Let $X^{(u,v)}$ be the subset of X on which $Y_{(u,v)}$ depends; that is $\pi(Y_{(u,v)}|X = x) = \pi(Y_{(u,v)}|X^{(u,v)} = x^{(u,v)})$, where $x^{(u,v)}$ is the values of the realisation x at the locations included in $X^{(u,v)}$. For example, in the case of motion blur, the set of pixels in the true scene which are included in $X^{(u,v)}$ would include those within a certain distance of the pixel (u, v) in the direction opposite to the motion. To model a point spread function, $X^{(u,v)}$ would include pixels in a neighbourhood around (u, v) echoing the shape of the point spread function.

In either the case of motion blur or a point spread function, the model for the data might then be

$$Y_{(u,v)}|x \sim \mathcal{N}(g_{(u,v)}(X), \sigma^2), \tag{3.3}$$

where $g_{(u,v)}(X)$ depends only on $X^{(u,v)}$.

In many situations g is linear, in which case we may write $Y_{(u,v)}|x \sim \mathcal{N}(Ax, \sigma^2)$, with A being a design matrix. This approach may be extended to an image which is not directly observed (i.e where there is no direct correspondence between elements of X and of Y). In that case, we no longer have data values $Y_{(u,v)}$ corresponding to particular pixels (u, v) in the true scene, and so we write $Y_i|x \sim \mathcal{N}(Ax, \sigma^2)$.

3.2.3 The posterior distribution of X given Y

Having chosen the likelihood and the prior, we can combine these to form the posterior distribution for the true scene, $\pi(x|y) \propto \pi(y|x)\pi(x)$. This is a joint distribution on the true scene, and may be used to make inference about X.

We shall now describe some methods which may be used to summarise the posterior distribution. These methods are in general applicable to any probability distribution.

Analysis of the posterior distribution

If X is defined on the pixel matrix, the posterior distribution is a function defined on a high dimensional (e.g. 10^5) space. We are usually interested in a point estimate as a summary statistic; this has a scalar value associated with each pixel and can therefore be presented in the form of an image.

One estimator is the value which maximises the posterior distribution. This is the maximum *a posteriori* (MAP) estimate of X. In some cases, where the model has been carefully chosen to be amenable, it is possible to find the MAP estimate by using deterministic numerical methods such as gradient ascent.

Iterated conditional modes (ICM) [2] gives an estimate of the true scene which is a local maximum of $\pi(x|y)$. At each iteration, the value of each pixel is updated to the maximum of its conditional distribution; that is, at iteration i we set

$$x^i_{(u,v)} = \arg\max_{x_{(u,v)}} \{\pi(x_{(u,v)}|x_{\setminus(u,v)}, y)\}.$$

The pixels are updated in turn, for example in a raster fashion. The algorithm is much less computationally demanding than sampling-based approaches, with convergence often occurring rapidly.

Another iterative algorithm which may be used to find a local maximum of $\pi(x|y)$ is expectation-maximisation (EM) [7][41]. At each iteration of EM, we perform an *expectation* step, where we derive an expression for the expectation of the log-posterior using the current values X. We then perform an *maximisation* step, where we calculate the value of X which maximises this expectation. The use of this algorithm for tomographic reconstruction will be discussed in Section 3.3.

Notwithstanding these methods of finding local maxima, the choice of model for which the MAP estimate can be efficiently calculated is relatively narrow, usually either placing severe restrictions on the allowed values of $x_{(u,v)}$ (e.g. binary images), or requiring both the likelihood and prior to be quadratic. If we wish to use a different model, or find statistics other than the MAP estimate (e.g. measures of spread), a sampling-based approach is required.

The usual method of choice, and the one that will be used extensively in this thesis, is Markov chain Monte Carlo (MCMC). In general, MCMC provides a way to draw a sample from a probability density (or mass function) $f(x)$. There are several books covering the theory and application of MCMC methods, including [15] and [32]. We shall use the Metropolis-Hastings algorithm [19], which is a method of constructing a Markov chain which has the target distribution, $f(x)$, as its stationary distribution. We shall describe the details of the implementation in each chapter where this method is used.

3.2.4 Non-Bayesian interpretation

In the above discussion, we have used a Bayesian formulation. If we wished to avoid using this paradigm, we could instead use a maximum likelihood approach. In this case, we would view the true scene X as the parameters to be inferred. For a directly observed image with no blur, we have as many data values as parameters, and with the normal likelihood model given above, the maximum likelihood estimator is $\hat{X} = y$. We may however use a penalised likelihood, with the penalty term used in much the same way as the Bayesian prior probability distribution—to encourage image characteristics such as local smoothness or similarity to a reference image. In this case, finding the MAP estimate can be recast as finding the penalised maximum likelihood estimate with an appropriate choice of penalty.

3.3 Statistical methods in tomography

We shall now turn to the specific application of statistical methods in tomographic reconstruction. As previously noted, many particular numerical algorithms can be interpreted as a regularised solution of a linear system, as the maximum of a

penalised likelihood, or as a MAP estimate. The phrase 'statistical reconstruction' has come to be used particularly for analyses where either the Poisson nature of the photon statistics is acknowledged [3], or where a Bayesian analysis is used.

One of the earliest applications of the maximum likelihood method in tomography was to emission tomography [35]. In this imaging modality, a radiopharmaceutical (a pharmaceutical product labelled with a radioactive tracer) is ingested, inhaled or injected into the patient. The spatial distribution of the activity within the patient follows that of the pharmaceutical, and may be characteristic of a particular disease state. The geometry of image acquisition is similar to that used in CT, except that the source of radiation is not an X-ray tube, but the radioactive material within the patient's body. The reconstruction task is to determine the spatial distribution of the tracer. The likelihood is Poisson, and an EM algorithm is derived to give the maximum likelihood estimate.

The method of [35] is not directly applicable to CT for a number of reasons. Firstly, the form of the likelihood is different. In emission tomography, it is the number of radioactive photons emitted from a point in the body which is a Poisson random variable, whose mean we wish to infer. In CT, if we assume that the beam is monenergetic, we are still dealing with a Poisson random variable. However, this represents the number of X-ray photons absorbed in the detector, whereas we wish to infer the attenuation properties of the patient. Secondly, the method of [35] uses an unpenalised likelihood (though penalised versions were subsequently developed e.g. [16]). The linear system in CT is typically ill-conditioned, and some form of regularisation is generally required to stabilise the solution. An iterative algorithm for calculating the maximum likelihood reconstruction for CT was given in [27]. Penalised likelihood approaches have also been used, for example in [10], [24] and [26].

An extension of the EM algorithm, ordered-subsets EM (OSEM) [23] was first applied to emission tomography data. The projections are partitioned into subsets. In one iteration of OSEM, one iteration of the EM algorithm is applied to each subset in turn, with the updated estimate being used as the starting condition for the next subset. Convergence is often faster than for EM. OSEM algorithms have subsequently been developed for CT [12].

The OSEM method is used in [11] to find the maximum likelihood reconstruction using a model which takes account of the energy dependence of X-ray attenuation. A two-component model is used, with pixels in the 2D reconstruction being soft tissue, bone, or a mixture of both, with the volume fraction being a parameter to be estimated. The energy dependence of the attenuation can be characterised for each of these tissues, avoiding problems with non-linearity.

Bayesian formulations have also been used for CT reconstruction. For example, the application to dental cone-beam CT using few projections is discussed in [25]. As in this thesis, a normal likelihood for the logarithm of the detector signal is used. The prior has two terms: a total variation term (penalising the sum of the reconstructed values) encourages the solution to have small areas of high voxel value with a low voxel value background. A term with the form of Equation 3.2 is also used, with Φ using the $\ell^{(1)}$ norm. A smooth approximation to the posterior distribution is used to enable efficient searching for the MAP estimate.

In [37], a Bayesian model is used with data from helical CT acquisition (where the patient is continuously translated through the bore of the scanner, whilst a fan-beam X-ray system is rotated around the central axis). The log-likelihood is approximated using a Taylor series expansion, and a novel convex prior (a further generalisation of a generalised Gaussian MRF) is introduced.

The EM, OSEM, and Bayesian reconstruction algorithms described above rely on the likelihood or posterior density being of such a form that an expression for the update at each iteration can be derived, or they achieve this by using an approximation. The focus is on creating a practical algorithm which converges rapidly without requiring an excessive use of RAM. The penalty terms used are chosen so that they provide regularisation by encouraging some degree of smoothness without sacrificing analytic amenability. In this thesis, we shall be less concerned with developing an efficient algorithm for reconstructing an arbitrary object from the projections. We shall focus on using Bayesian models to describe high atomic number fiducial markers implanted in soft tissue, and use MCMC to sample from posterior distributions.

CHAPTER 4

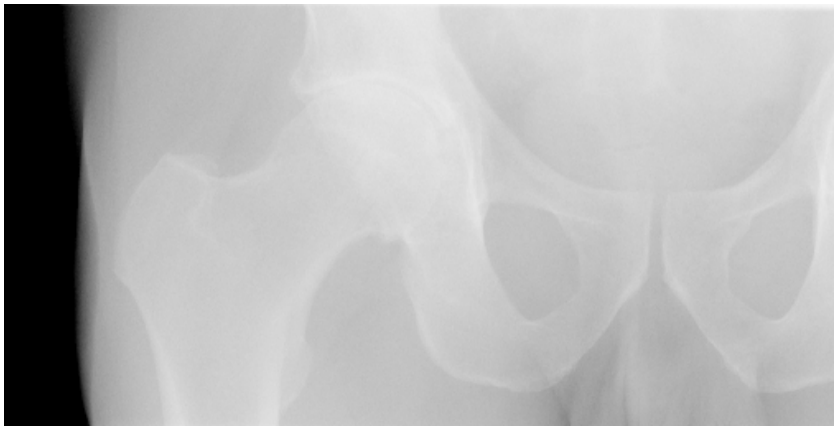
STATISTICAL ANALYSIS AS A PRE-PROCESSING STEP

4.1 Introduction

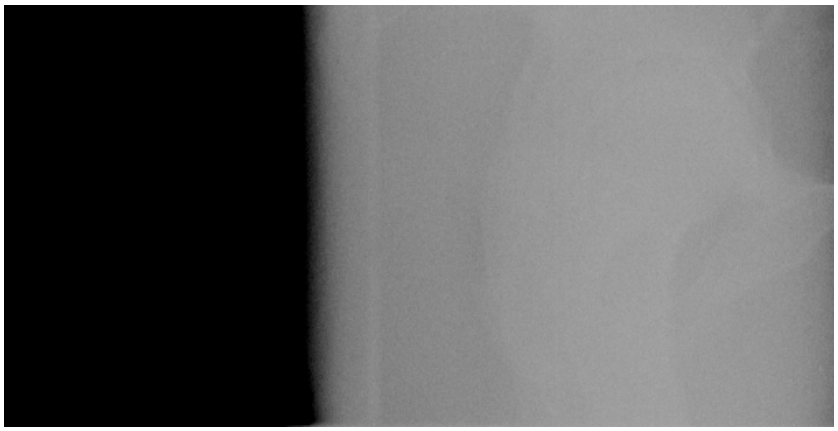
In this chapter, we shall fit a statistical model to image data, as a practical demonstration of many of the ideas discussed in Chapters 2 and 3. The data we shall use are of a male pelvis, but without any implanted fiducial markers—the 'no markers' dataset. We shall consider each projection image in isolation, two of which are shown in Fig. 4.1.

We shall define the true scene to be the 'ideal' pixel values in these images, i.e. the pixel values that would be obtained in the absence of noise. For each projection angle, we shall estimate the 2D true scene from the noisy projection image.

These estimates of the true projection image at each angle will be used in place of the noisy images in a Feldkamp-Davis-Kress (FDK) reconstruction. The resulting voxellised representations of the patient will be compared with those obtained by applying the standard technique for removing noise: frequency-domain filtering of the 2D projection data.



a: Anterio-posterior projection.



b: Lateral projection.

Figure 4.1: Two projection images.

This work is similar to [20], where statistical methods are used with emission tomography data. The authors found a maximum likelihood estimate of the true scene, then used filtered back projection to obtain a 3D reconstruction.

4.2 Noise reduction methods

In Chapter 2, the filtered back projection algorithm for computed tomographic reconstruction was described as a method of generating a reconstruction of an object from projections through that object. The 'filtering' in filtered back projection

refers to the application of a ramp filter $g_\infty(\tau)$. We remarked that the ramp filter is a high pass filter: it attenuates signals such that the zero-frequency component is removed entirely, and the amount of attenuation decreases as spatial frequency increases. After applying the ramp filter, a reconstruction is created by back-projecting the filtered data. In the continuous case with 1D projections, this creates a perfect reconstruction (although with a finite number of pixellised projections in 3D there are some errors due to discretisation and due to the inadequacy of the image acquisition geometry to provide projections filling the 3D Radon space).

The ramp filter is noise-enhancing—it leaves the highest spatial frequencies (where we expect to find uncorrelated noise) unattenuated. It is usual to apply a noise-reducing low-pass filter to the 2D projection data, in addition to the noise-enhancing ramp filter. A typical example of a low-pass filter is the Hamming filter, defined as

$$g_{\text{Hamming}}(\tau) = 0.54 + 0.46 \times \cos(\pi\tau/\tau_N) \tag{4.1}$$

where τ is the spatial frequency and τ_N is the Nyquist frequency for the system. This filter is shown in Fig. 4.2, along with the ramp filter. Also shown is the compound filter which results from applying both: at each spatial frequency, the value of the compound filter is the product of the coefficients of the Hamming and ramp filters. Fig. 4.3 shows the effect of applying the various filters to one image.

Fig. 4.4 shows a transaxial slice through a reconstruction obtained using the FDK algorithm, where no smoothing was used on the projection images before reconstruction. There is a lot of speckle in the image, and the edges of the bones are hard to visualise. Fig. 4.5 shows the same transaxial slice as in Fig. 4.4, but from a reconstruction where a Hamming filter was applied in addition to the ramp filter. The image is improved, with less speckle and the edges of the bones more clearly defined. Some soft tissue contrast is also seen.

This demonstrates that applying some spatial smoothing to the 2D projection data before back projecting improves the reconstruction. However, we do not wish to over-smooth, lest we lose resolution. Using a low-pass filter applies smoothing equally to sharp edges and to areas where there are no large-scale features.

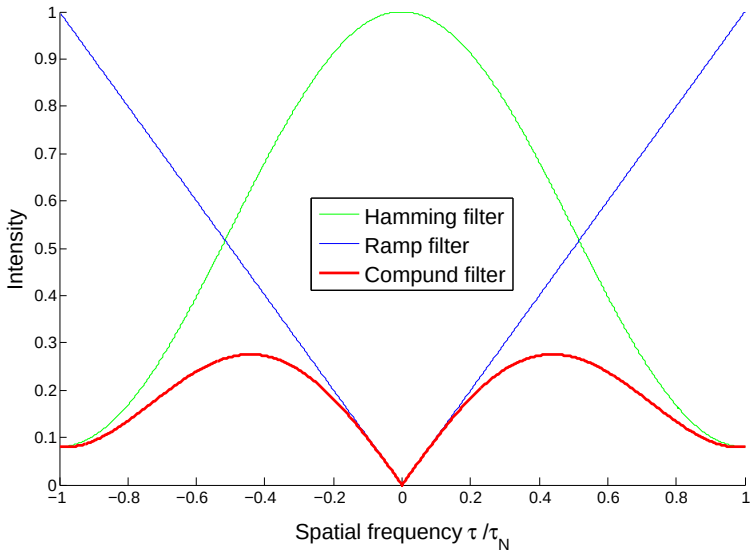


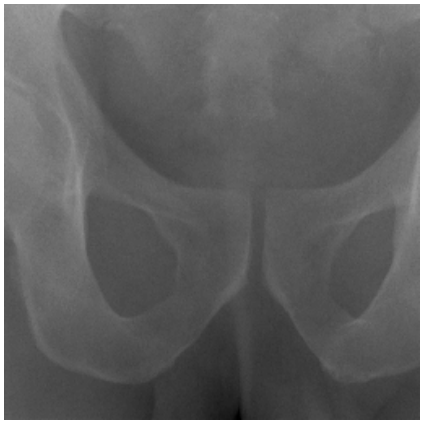
Figure 4.2: Graphical representation of filters.

The three filters shown are the Hamming filter (low-pass), the ramp filter (high-pass), and the compound filter that results from applying both of these together.

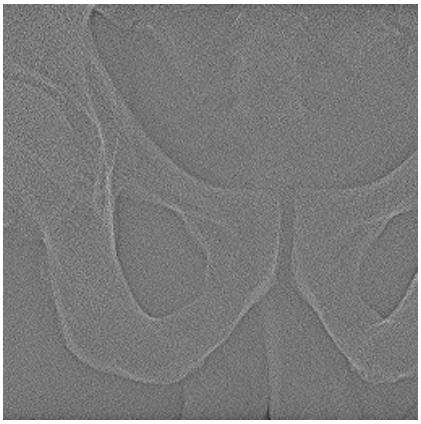
Instead of using a low-pass filter, we shall use a Bayesian statistical model for the 2D projection data in which the prior encourages smoothness. In doing so, we hope to recover an estimated image in which the measurement noise is suppressed, but where the sharp edges of anatomical features are preserved.

4.3 Statistical model

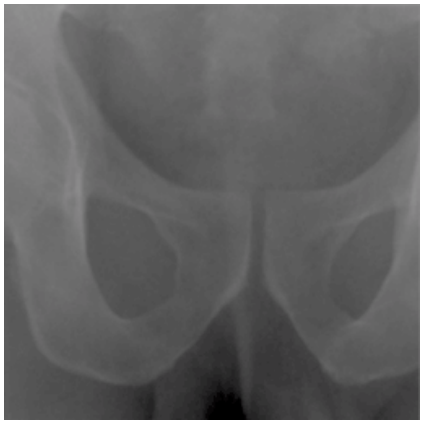
In this section, we shall define our statistical model. We shall denote the true scene by X, with the value of pixel i in the true scene be denoted by X_i. We denote the measured projection data by Y. We consider the image to be directly observed; that is, the data Y are measured on the same Cartesian pixel grid as the true scene X.



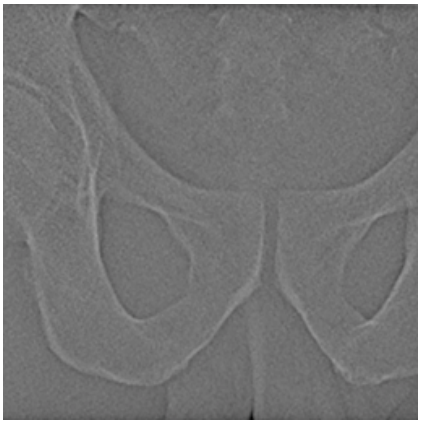
a: Original image.



b: Ramp filter.



c: Hamming filter.



d: Compound filter.

Figure 4.3: Filters applied to an image.

The images are filtered with the ramp filter, Hamming filter, and the compound filter created by applying both together.

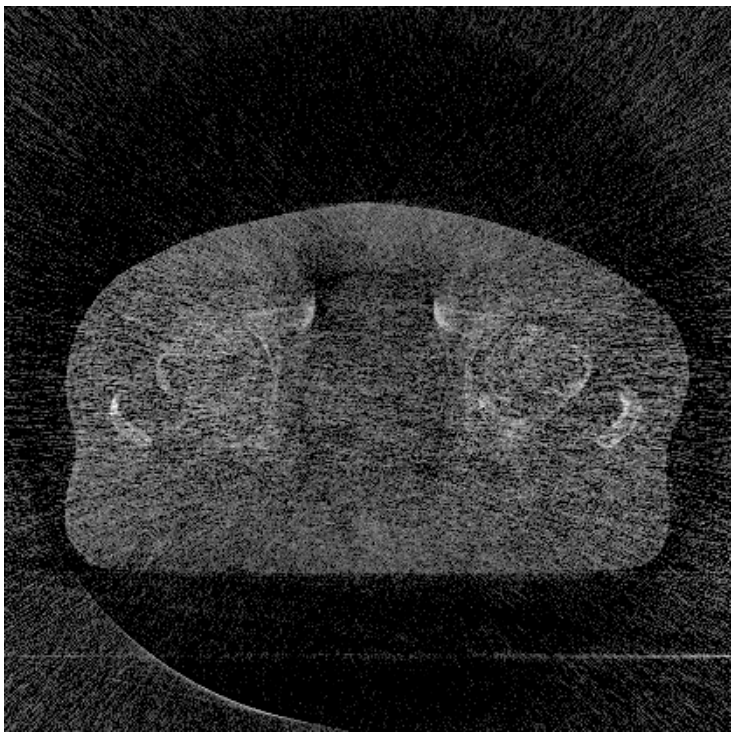


Figure 4.4: Reconstruction with no noise reduction.

A transaxial slice through an FDK reconstruction. No noise reduction was applied to the projection images before reconstruction.

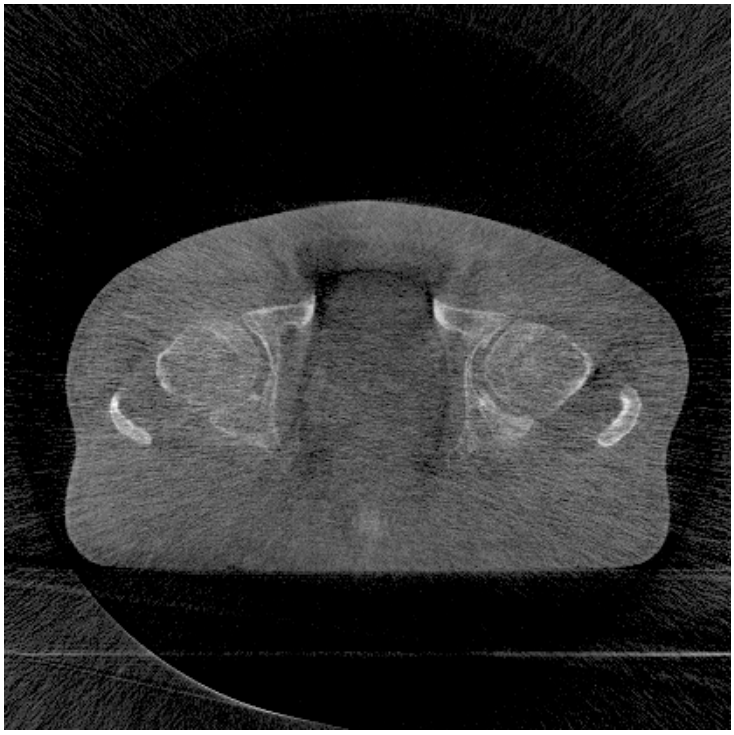


Figure 4.5: Reconstruction using a Hamming filter.

A transaxial slice through an FDK reconstruction. A Hamming filter was applied to each Fourier-transformed projection image, in addition to the ramp filter.

4.3.1 Prior

For this analysis, the true scene in each 2D projection was taken to be a Markov random field. We used a first order neighbourhood (so each pixel not at the edge of the image has four neighbours).

Following the notation used in the previous chapter, we define the joint probability distribution for X as

$$\pi(x) \propto \exp\left\{ -\sum_{c \in \mathfrak{C}} \Phi_c(x) \right\} \tag{4.2}$$

with the clique potential

$$\Phi_c(x) = \beta \sum_{i,j \in c} |x_j - x_i|. \tag{4.3}$$

The conditional probability that pixel i takes value x_i, given the values of its neighbours, is then

$$\pi(x_i | x_{\setminus i}) = \frac{1}{Z} \exp\left\{ -\beta \sum_{j \in \delta(i)} |x_j - x_i| \right\}. \tag{4.4}$$

The value of β determines the strength of the correlation between neighbouring pixels. Results using different values for β will be given in Section 4.4. Z^{-1} is a normalising constant, whose value is unknown. However, we shall use MCMC (where only the ratio of the prior probabilities of two different true scenes will be required) to sample from the posterior distribution.

We note that this prior is improper. We could make it integrable, for example by placing restrictions on the range of allowed values of the $x_{(u,v)}$ or defining a proper distribution for one of the pixel values. We do not do this, relying instead on the data to locate the mass of the posterior distribution, which is proper.

4.3.2 Likelihood

The data values Y_i were assumed to be conditionally independent, given the true scene X. We used a normal model, with

$$Y_i|x_i \sim \mathcal{N}(x_i, \sigma^2). \tag{4.5}$$

As mentioned in Chapter 2, images of Perspex blocks (which mimic soft tissue) were acquired. The variance was determined from the Perspex images; a value of $\sigma^2 = 500^2$ was used.

4.3.3 Posterior

The posterior distribution $\pi(x|y)$ is proportional to the product of the prior distribution for x and the likelihood of y given x. We wish to find an estimate of the true scene, which we can use in an FDK reconstruction. We shall use the marginal posterior mean (MPM) value for each pixel. We shall refer to the set of MPM values as the MPM image for each projection, and we shall estimate this using an MCMC sampler.

MCMC sampling

We used a Metropolis-Hastings MCMC sampler, updating each pixel in turn in a raster fashion. Let the value for pixel i at iteration t be $x_i^{(t)}$, with t incremented after a complete cycle through all the pixels. We use a normal proposal, with the proposed value being $x_i^\star \sim \mathcal{N}(x_i^{(t)}, \sigma^2_{\text{prop}})$. The effect of different choices of σ^2_{prop} will be demonstrated in Section 4.4.

In each MCMC run, the sampler was initialised with the values from the measured data for that projection. The MCMC sampler was run on each projection for 1100 iterations, with the first 100 discarded as burn-in. These are relatively short runs; longer runs were performed in some cases but did not produce a visually detectable difference in the subsequent FDK reconstructions.

4.4 Experimental results

4.4.1 Parameter choices

Choice of smoothing parameter, β

In the prior distribution, we can choose a value for the parameter β, which governs the strength of the interactions between neighbouring pixels. This is demonstrated in Fig. 4.6 for a small region of an anterio-posterior projection containing the femoral head. The four images are MPM images produced by sampling from distributions with different values for β—the higher the value of β, the smoother the image. To determine the optimal choice of β, FDK reconstructions were carried out using MPM images with a range of different β values, using a number of different data sets including pelvis and other anatomical sites (which we have not included here). The value $\beta = 0.002$ gave a good reconstruction in each case, providing a balance between effective noise reduction compared to the original image data, whilst preserving the edges to provide definition around bony anatomy.

Choice of variance for proposal distribution, σ^2_{prop}

We want to sample efficiently from the posterior distribution. We can tune the Metropolis-Hastings sampler by changing the variance in the proposal distribution σ^2_{prop}. If a small value of σ^2_{prop} is used, the proposed value will be very similar to the current value. The ratio of posterior densities is therefore likely to be close to one, and most moves will be accepted. However, the sampler will only move a very small distance around the space of possible configurations at each move. If a large values of σ^2_{prop} is used, each proposed move will be more likely to take the sampler to an area of the target distribution far from the current value. However, many of the proposed values will have much lower probability, and will therefore be unlikely to be accepted. The chain becomes 'sticky', staying in one configuration for many iterations. The acceptance rate (the percentage of proposed moves which are accepted) is a convenient way to assess how efficiently the chain is exploring the distribution. An acceptance rate of 23% is demonstrated in [33] to be fairly widely applicable in practice.

a: $\beta = 0.0002$

b: $\beta = 0.002$

c: $\beta = 0.02$

d: $\beta = 0.2$

Figure 4.6: Different levels of smoothing in MPM images.

A small region of an anterio-posterior image, showing the femoral head. Each is the MPM image, with different values of β (as defined in Equation 4.4).

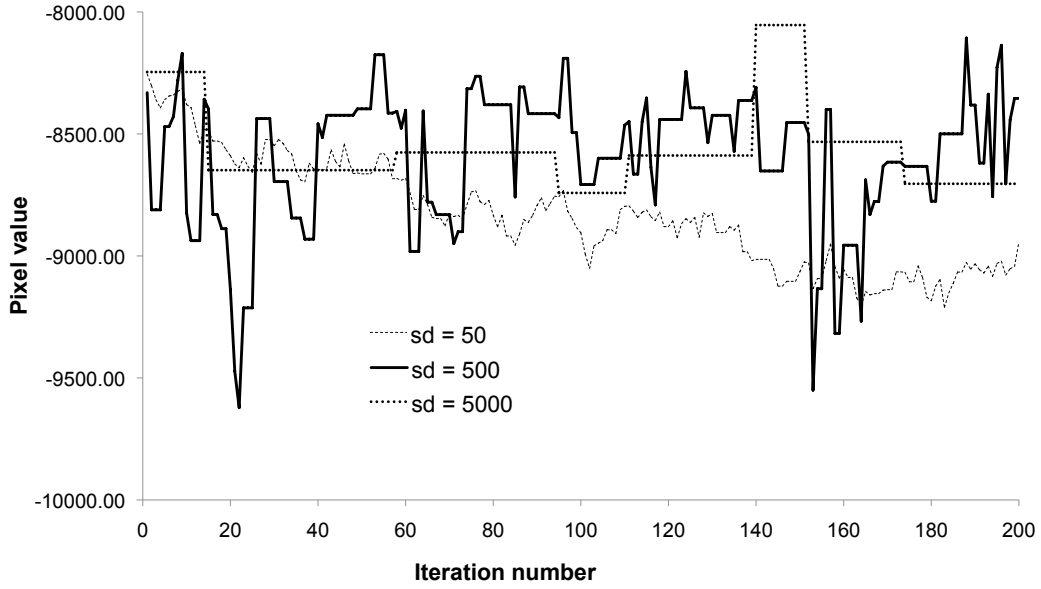


Figure 4.7: The effect of changing the proposal standard deviation.

Traces of the trajectory of the value of one pixel. The three traces are from samplers with different values of the proposal standard deviation.

Fig. 4.7 shows the trajectory of the value of one pixel in three different samplers, each with a different value of σ^2_{prop}. This demonstrates one chain which is making very small steps ($\sigma^2_{\text{prop}} = 50$), one which is exploring the posterior distribution reasonably efficiently ($\sigma^2_{\text{prop}} = 500$), and one which is not moving freely ($\sigma^2_{\text{prop}} = 5000$). The acceptance rates were 89% for $\sigma^2_{\text{prop}} = 50$, 42% for $\sigma^2_{\text{prop}} = 500$, and 4% for $\sigma^2_{\text{prop}} = 5000$. The value of $\sigma^2_{\text{prop}} = 500$ was chosen.

4.4.2 Smoothing of projection images

Fig. 4.8 shows three images of the femoral head: the original image, the image after a Hamming filter has been applied, and an estimate of the MPM image (obtained from an MCMC run). The Hamming-filtered and MPM images are less noisy than the original. Fig. 4.9 shows a profile of the pixel values across part of the image, again at the level of the femoral heads. This also demonstrates the smoothing effect. The MPM image is smoother than the Hamming-filtered image where there

is localised noise, but it retains the sharpness in the large changes where there are edges.

4.4.3 Reconstructions

The FDK reconstructions in Figures 4.4 and 4.5 were carried out using the complete data, consisting of 626 projection images. For the purposes of the following comparison, these may be considered to be baseline images. With the full data set, there was little visible difference in the reconstructions using the MPM and Hamming-filtered images. Differences became visible when a subset of the images were used. The results shown here are reconstructions using a subset consisting of 79 projection images (one in eight). Achieving a good reconstruction with fewer images would allow less radiation to be used in the image acquisition, reducing the risk of causing a secondary cancer.

Fig. 4.10 shows a transaxial slice from the FDK reconstruction obtained using this subset of projections, without applying any smoothing filter. Fig. 4.11 shows the same transaxial slice from the FDK reconstructions obtained using the Hamming-filtered images and the estimates of the MPM images.

4.5 Discussion

In Fig. 4.11, the differentiation between the bone and soft tissue is better in the reconstruction from MPM images than in the one using the same number of Hamming-filtered images. The MPM image is smoother than the original data, especially in areas where there are no large-scale features. In order for a low-pass filter to have equivalent smoothing in these areas, the edges would be unacceptably blurred.

The analysis presented here has demonstrated some of the ideas presented in the introductory chapters, showing the value of using a statistical analysis for each of the projection images. This acts as a pre-processing step before applying a ramp filter, and is used in place of the usual low-pass filter. However, we have not addressed the problems caused by the fiducial markers. Fig. 4.12 shows a reconstruction from MPM images, again using one in eight of the images, but from

a: Original image.

b: Hamming-filtered image.

c: Estimate of MPM image.

Figure 4.8: The effect of different processing.
The region shown is the same as in Fig. 4.6, processed in different ways.

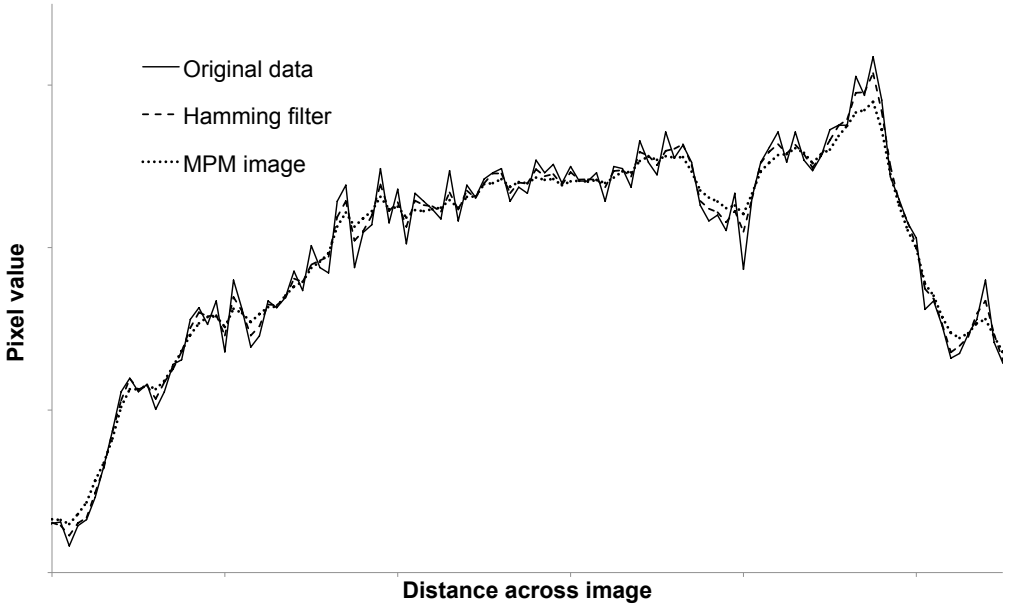


Figure 4.9: Profiles of pixel values in processed images.

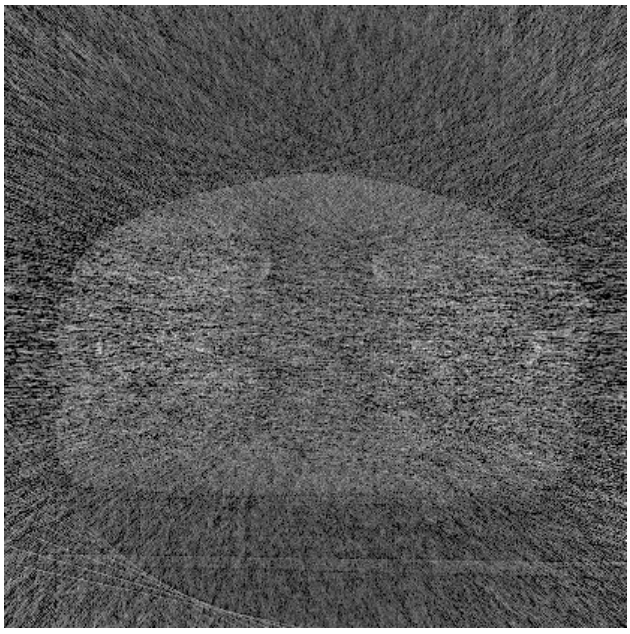
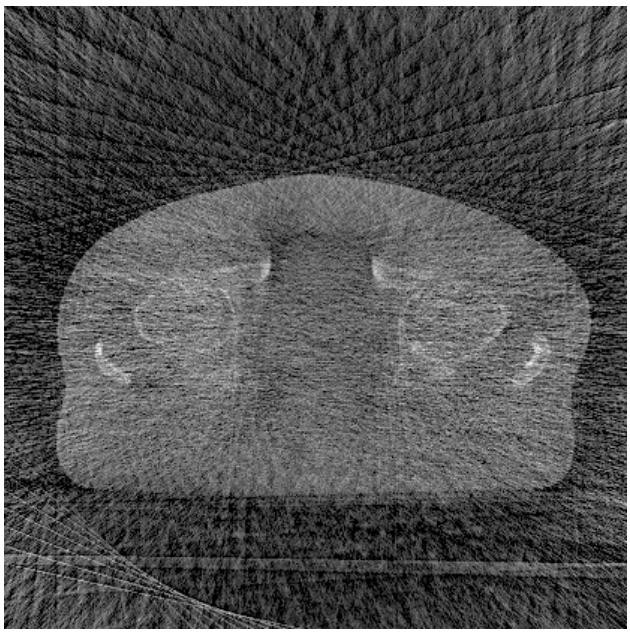
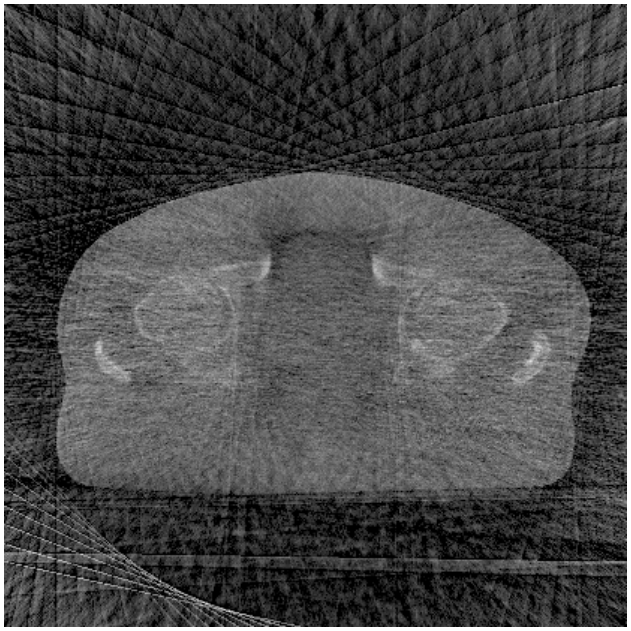


Figure 4.10: Reconstruction using 79 original images.



a: Reconstruction from Hamming-filtered images.



b: Reconstruction from estimates of MPM images.

Figure 4.11: Reconstructions using 79 processed images.

Transaxial slices from FDK reconstructions using 79 projection images, processed in different ways before carrying out the reconstruction.

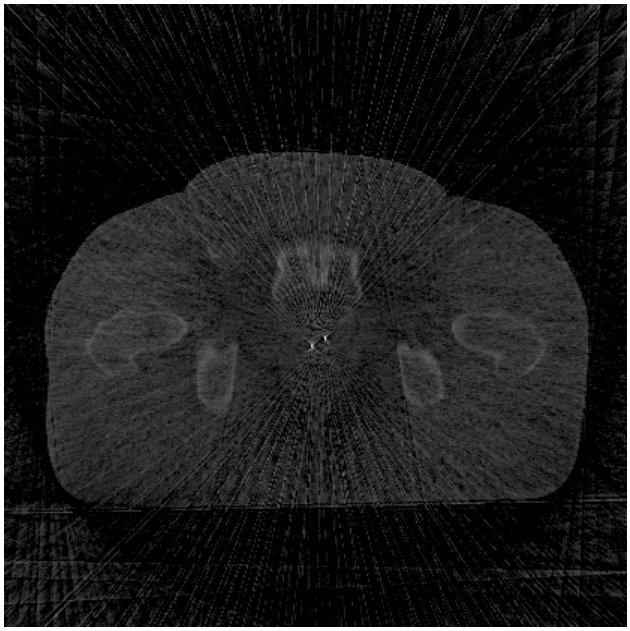


Figure 4.12: Reconstruction of patient with markers implanted.

Transaxial slice from an FDK reconstruction of a patient with markers implanted, using 79 MPM images.

a patient with implanted markers. The artefacts they cause have not been reduced by carrying out the statistical analysis presented in this chapter.

As discussed in Chapter 2, the markers cause artefacts because, in carrying out an FDK reconstruction, we are finding a solution to a linear system. The system is not linear, especially when gold markers are included. In order to reduce the streak and star artefacts, we need to model the markers as a special case. This will be the subject of the next two chapters.

CHAPTER 5

MODELLING MARKERS IN 2D PROJECTIONS

5.1 Introduction

In the previous chapter, we illustrated the methods described in Chapters 2 and 3 by using statistical methods on the 2D projection data. We fitted a Bayesian model, using the estimates of the marginal posterior mean (MPM) projection to reconstruct a 3D volume with the Feldkamp-Davis-Kress (FDK) algorithm. We did not model the fiducial markers in any particular way, making no attempt to find their location nor to remove the beam hardening artefacts that they cause.

In this chapter, we shall again use a Bayesian model for the 2D projection data. Within each projection, we shall specify a stochastic model for the 'true scene', i.e. the pixelised projection image that would be acquired in the absence of noise. Rather than try to describe the physical gold marker, we parameterise projected markers in terms of the pixels where the markers appear in this true scene. For the purposes of this chapter, we shall use the word 'marker' to mean this pixelised representation of a projected marker, rather than the physical entity implanted in the patient. We shall specify a prior distribution for these markers and a model for the data given a marker, and use Markov chain Monte Carlo (MCMC)

sampling methods to locate the marker end points. We shall combine the posterior probabilities for each projection in real 3D space.

We start by selecting small regions in the 2D projection image where we believe the markers may be by using a morphological analysis (described in Section 5.2). The model for the markers within these regions will be defined in Section 5.3. We shall use MCMC in a novel way to locate the marker ends, as described in Section 5.4. The results of the 2D sampling may be used in two ways. Firstly, in Section 5.6, we use them to assign to points in 3D the probability that there is a marker end at each of these points. Secondly, we describe in Section 5.7 how the projection images may be processed in order to reduce or remove the artefacts. Both uses of the sampling will be demonstrated using patient data.

The removal of artefacts by processing the projection data is also the goal of [30] and [42]. In the former paper, the location of the metallic objects is determined by thresholding an initial reconstruction. The location of the object in each projection is calculated. The values of the pixels representing the object are replaced by values obtained by fitting a polynomial surface to the surrounding pixel values, with added Gaussian noise. In the latter paper, fiducial markers and other metallic objects are removed from cone-beam reconstructions by prompting the user to identify the metal objects on two projections. A thresholding method is used to identify the extent of the metal object, which is then tracked through the other projections. The metal objects are removed in each projection and replaced by pixel values chosen to minimise a variational cost function, based on the pixel values in the area surrounding the object. In both papers, 3D reconstructions are obtained by using the 'corrected' projections.

The markers used in [30] and [42] are of the 'seed' type (solid gold), with very high contrast, so that locating the markers using methods such as thresholding is possible. The markers depicted in this thesis are coils of fine gold wire, with lower contrast, which precludes detecting the markers in the projection images by thresholding. The primary objective of [30] and [42] is artefact removal, with neither paper addressing the question of the uncertainty in the location and only one finding the markers without user interaction. Our aims are to find an estimate (with uncertainty) of the marker location, as well as to remove the artefacts.

Unless otherwise stated, the computer programs for this chapter were written in Python[1], with Pymorph[8] used for the morphological analysis.

5.2 Morphological analysis

In Section 5.3, we shall describe our model for the markers. In order to make the analysis of this model feasible, we start by using morphological processing on each of the 2D images to narrow the search space to a number of regions of interest (ROIs).

The analysis used was a three-stage process, the first two of which were accomplished using the morphological operations. Firstly, the variation in background pixel value was removed, effectively leaving high-frequency detail; secondly, features which appeared as straight lines were identified; finally, regions of interest including clusters of candidate features were identified.

We shall now give a brief introduction to the basic morphological operations (further information can be found in [8]), before describing each of the three steps. The analysis was restricted to a region covering the centre of the patient. This was 256×256 pixels, centred at the point at which the central axis of the X-ray beam intersects the detector (i.e. centred in the patient cranio-caudal direction, and at the right of the detector in the patient anterio-posterior/right-left direction).

5.2.1 Basic operations: dilation, erosion, opening and closing

The basic operations used in morphological image analysis are dilation and erosion. We shall first describe and illustrate these for a 2D binary pixelised image, then generalise to a greyscale image.

Dilation of a binary image

Let A be a binary image, i.e. a mapping from integer indices (u, v) to $\{0, 1\}$. Let the *structuring element* B be another binary image. We define $\mathcal{S}_B$ to be the support of B, i.e. those values of (u, v) for which the mapping B is defined, and require

[1] `http://www.python.org`

the origin $(0,0) \in \mathcal{S}_B$. We write $A^{(1)}$ to be the set $\{(u,v)$ s.t. $A(u,v) = 1\}$, with corresponding definitions for $B^{(1)}$ and $C^{(1)}$.

The *dilation* of A with B, denoted $A \oplus B$, is the binary image C such that

$$C^{(1)} = \bigcup_{(u,v)\in A^{(1)}} \{(u+i, v+j) \text{ s.t. } (i,j) \in B^{(1)}\}.$$

That is, we move the origin of the structuring element B to each non-zero pixel in A and wherever the value of B is non-zero, we assign C the value 1. This is illustrated in Fig. 5.1.

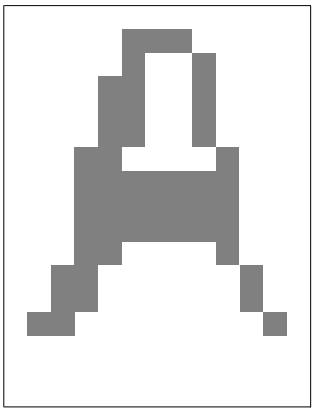
Note that there is another convention for defining a binary image, namely the set of non-zero elements (i.e. what we are calling $A^{(1)}$). Our choice of definition (the binary image including both the zero and non-zero values) extends more naturally to greyscale images.

Erosion of a binary image

Using the same definitions as above, the *erosion* of A with B, denoted $A \ominus B$, is the binary image C such that

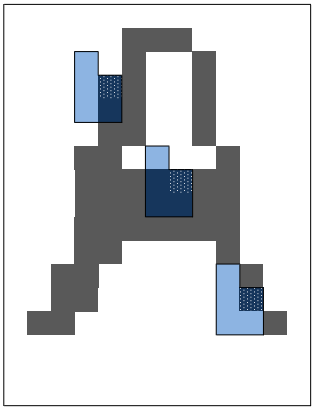
$$C^{(1)} = \{(u,v) \text{ s.t. } (u+i, v+j) \in A^{(1)} \; \forall (i,j) \in B^{(1)}\}.$$

That is, we move the origin of the structuring element B to each non-zero pixel in A. Whenever all the non-zero pixels in B align with non-zero pixels in A, we assign the value 1 to the pixel in C corresponding to the origin of B. This is illustrated in Fig. 5.2.

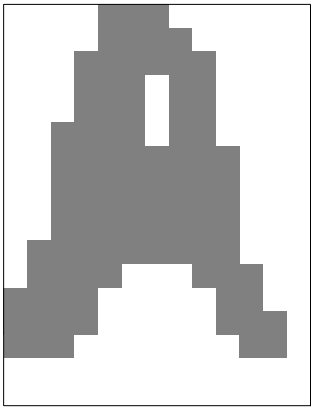


a: The binary image A.

b: The structuring element B.



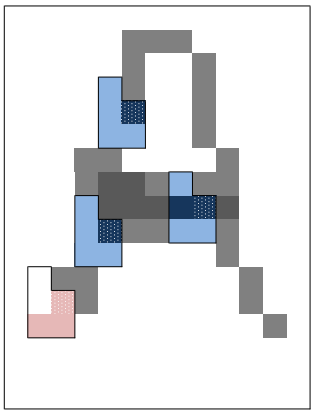
c: Calculating the dilation.



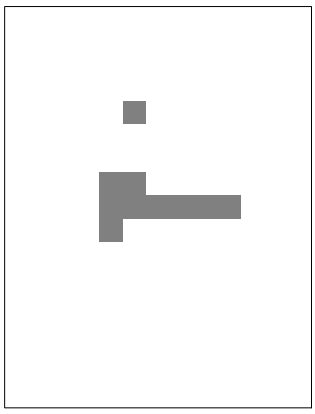
d: The dilation of A with B, $A \oplus B$.

Figure 5.1: Illustration of dilation.

In Figures 5.1a and 5.1b, the original image and structuring elements represent the letters 'A' and 'b' respectively. The origin of the structuring element, $(0, 0)$, is highlighted by dots. In Fig. 5.1c, the image A in shown in dark grey / dark blue, with three of the structuring element positions highlighted. The pixels in blue are in $B^{(1)}$ when the origin of B is placed on a pixel in A. The union of these blue pixels, with the origin placed over all the pixels in A, forms the dilation of A with B.



a: Calculating the erosion.



b: The erosion of A with B, $A \ominus B$.

Figure 5.2: Illustration of erosion.

The image and structuring element are shown in Figures 5.1a and 5.1b. In 5.2a, the original image A consists of the pixels shown in anything other than white. Three structuring element positions are shown in blue; in each of these cases, the whole of $B^{(1)}$ lies within $A^{(1)}$ when the origin of B is placed at the points shown with dots. Also shown is another structuring element position (highlighted in pink) where $B^{(1)}$ is not wholly contained in $A^{(1)}$. The erosion consists of all the points at which we can place the origin of B with the result that $B^{(1)}$ lies within $A^{(1)}$.

Opening and closing of a binary image

The operations of dilation and erosion can be combined in two further operations: the closing and the opening of a binary image by a structuring element. The closing of A with B is defined as

$$A \bullet B = (A \oplus B) \ominus B,$$

that is, we calculate the dilation of A with B, then erode with B. This is illustrated in Fig. 5.3. The closing may be thought of as 'filling in' any holes in A which are smaller than B.

The opening of A with B is defined as

$$A \circ B = (A \ominus B) \oplus B,$$

that is, we calculate the erosion of A with B, then dilate with B. This is also illustrated in Fig. 5.3. Intuitively, the opening may be thought of as removing any parts of A which are smaller than B, taking into account the shape of B.

Modification for greyscale images

So far in this section, we have only considered binary images, where each pixel can take on the values 0 or 1. We shall now describe how these concepts may be adapted for greyscale images, where each pixel can take values in a larger set (which might be finite, countable, or uncountable). We shall consider images on a finite grid, indexed by (u, v). We shall use the convention of displaying low pixel values as dark and high pixel values as bright (note that this is the converse to the binary images above, where we displayed the zero-valued pixels as white and the one-valued pixels as grey).

Let $A(u, v)$ be the value of the image A at pixel (u, v). The structuring element may also take on greyscale values, and we write $B(r, s)$ for the value of B at (r, s) for $(r, s) \in \mathcal{S}_B$, the support of B. The dilation of A with B is then defined as

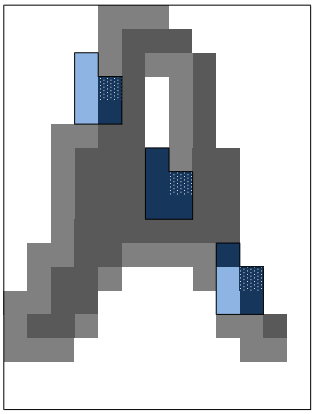
$$[A \oplus B](u, v) = \max_{(r,s)\in\mathcal{S}_B} \big\{A(u + r, v + s) + B(r, s)\big\}.$$

In words, we place the origin of the structuring element B over pixel (u, v) in the image A, add the value of B to A at each pixel, and then find the maximum value of these sums. We assign this maximum to be the pixel value of $A \oplus B$ at pixel (u, v).

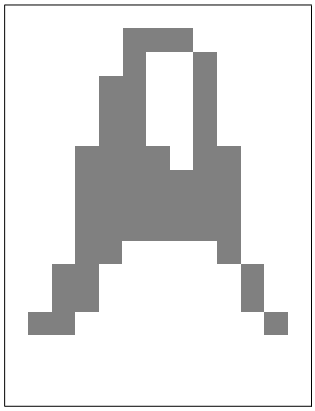
The erosion of a greyscale image A with structuring element B is defined as

$$[A \ominus B](u, v) = \min_{(r,s)\in\mathcal{S}_B} \big\{A(u + r, v + s) - B(r, s)\big\}.$$

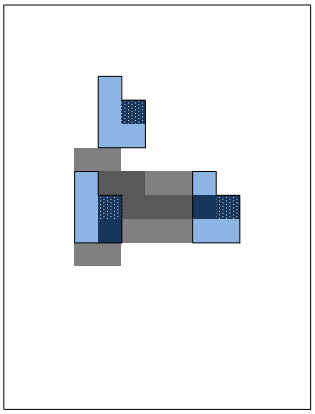
In this case, we place the origin of B at pixel (u, v), then subtract the pixel value of B from the pixel values in A. We assign the minimum of these differences to be the pixel value of $A \ominus B$ at pixel (u, v). The dilation and erosion of greyscale images are illustrated using part of one of the projection images in Figures 5.4 and 5.5.



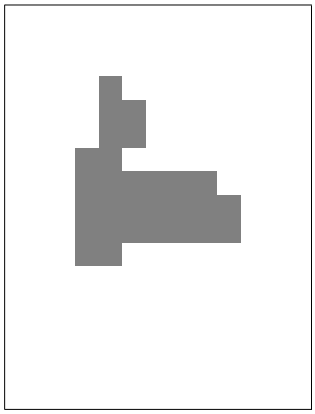
a: Calculating the closing.



b: The closing of A with B, $A \bullet B$.



c: Calculating the opening.



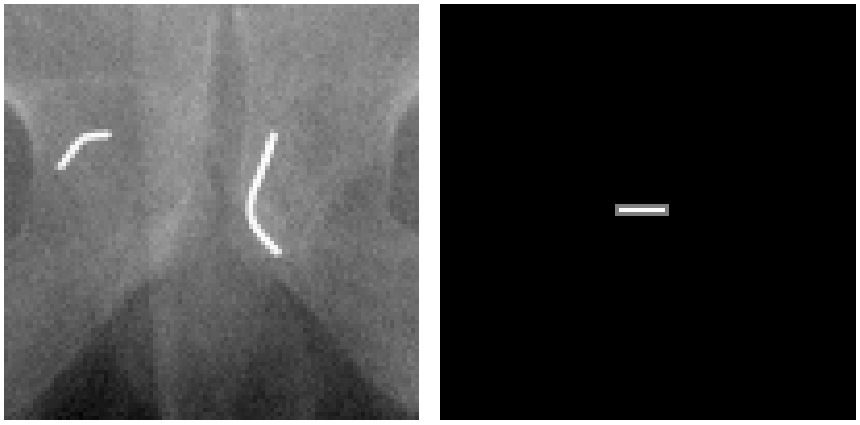
d: The opening of A with B, $A \circ B$.

Figure 5.3: Illustration of closing and opening.

The image and structuring element are the same as in Fig. 5.1.

In 5.3a, the dilation of A with B ($A \oplus B$) consists of all the non-white pixels (c.f. Fig. 5.1d). Three position for the structuring element are shown in blue; in each case, $B^{(1)}$ is contained wholly inside $A \ominus B$. The dark grey and dark blue pixels indicate all the positions for the structuring element where this is the case; these form the closing.

In 5.3c, the erosion of A with B ($A \ominus B$) is shown in dark grey / dark blue (c.f. Fig. 5.2b). Three positions for the structuring element are shown; in each case, the origin of the structuring element is in $A \ominus B$. The union over all such positions of the pixels which are in $B^{(1)}$ forms the opening.



a: Original image A. **b:** Structuring element B.

Figure 5.4: Original greyscale image and structuring element.

The original image is part of a projection. The structuring element is of size 3×12 pixels. The black area is outside the support of B, with the values in B being 1 (grey) and 2 (white). Morphological transformations using this image and structuring element are shown in Figures 5.5 and 5.6.

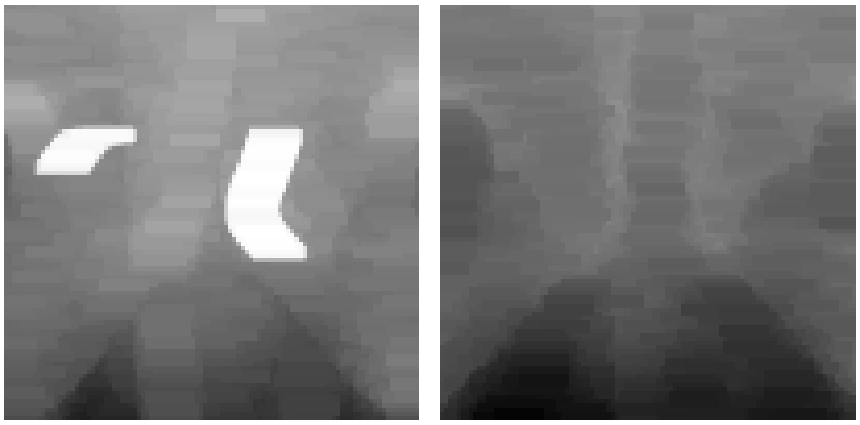
The opening and closing of A with B are defined in the same way as for binary images. That is, the opening

$$A \circ B = (A \ominus B) \oplus B$$

and the closing

$$A \bullet B = (A \oplus B) \ominus B.$$

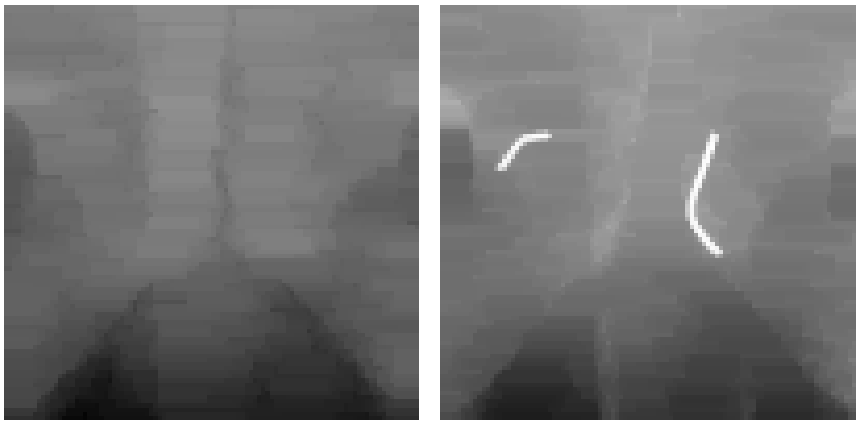
The opening has the effect of removing high-valued (i.e. bright) features which are smaller than the structuring element, whereas the closing fills in low-valued (i.e. dark) areas. These operations are illustrated in Fig. 5.6 (again using the image and structuring element in Fig. 5.4). In the opening, bright features smaller than B (notably the markers) have been removed. In the closing, dark areas smaller than B have been filled in (such as the dark feature running from the top to the centre of the image).



a: The dilation of A with B, $A \oplus B$.

b: The erosion of A with B, $A \ominus B$.

Figure 5.5: Illustration of dilation and erosion on a greyscale image. The original image and structuring element are illustrated in Fig. 5.4.



a: The opening of A with B, $A \circ B$.

b: The closing of A with B, $A \bullet B$.

Figure 5.6: Illustration of the closing and opening of a greyscale image. The original image and structuring element are illustrated in Fig. 5.4.

				u			
	6	9	12	16	12	9	6
	9	12	16	19	16	12	9
	12	16	19	22	19	16	12
v	16	19	22	25	22	19	16
	12	16	19	22	19	16	12
	9	12	16	19	16	12	9
	6	9	12	16	12	9	6

Table 5.1: The values of the 'ball' structuring element.
This structuring element is used in the top-hat transform.

5.2.2 Application to projection images

Now that we have defined and illustrated the basic morphological operations, we shall describe how we have applied them to find markers in 2D projection images. For each image, we shall identify ROIs which have a high probability of containing one or more markers. The morphological analysis has three stages: a top-hat transform, detection of linear features, and ROI selection.

Removing the variation in background: the top-hat transform

The first stage of the morphological processing is the removal of large-scale variation in pixel value. For this, we use the top-hat transform.

The top-hat transform is defined as the difference between the original image and its opening with a particular structuring element. We used a 'ball' structuring element (an integer approximation to that created in Matlab using `strel('ball',3,25)`), shown in Table 5.1 and Fig. 5.7. Fig. 5.8 shows the results of using this top-hat transform on three projection images.

Detection of linear features

For the second stage of the morphological analysis, we wish to detect linear features. We require structuring elements whose shape matches the features we wish to detect, so we create structuring elements representing linear features of length five pixels, as illustrated in Fig. 5.9. These detect features where there are bright pixels arranged in one of the configurations represented by a structuring

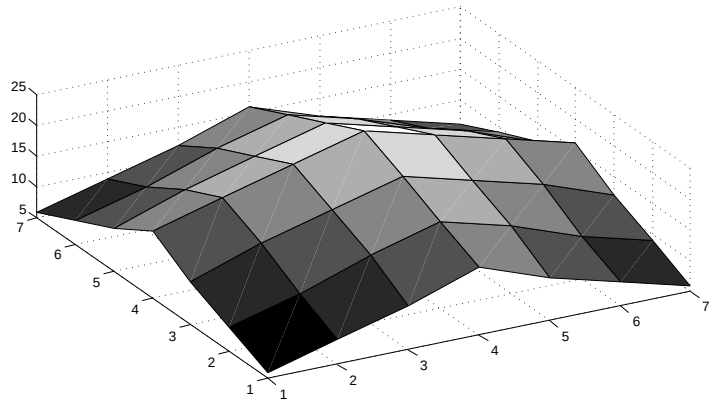


Figure 5.7: Surface plot of the 'ball' structuring element.

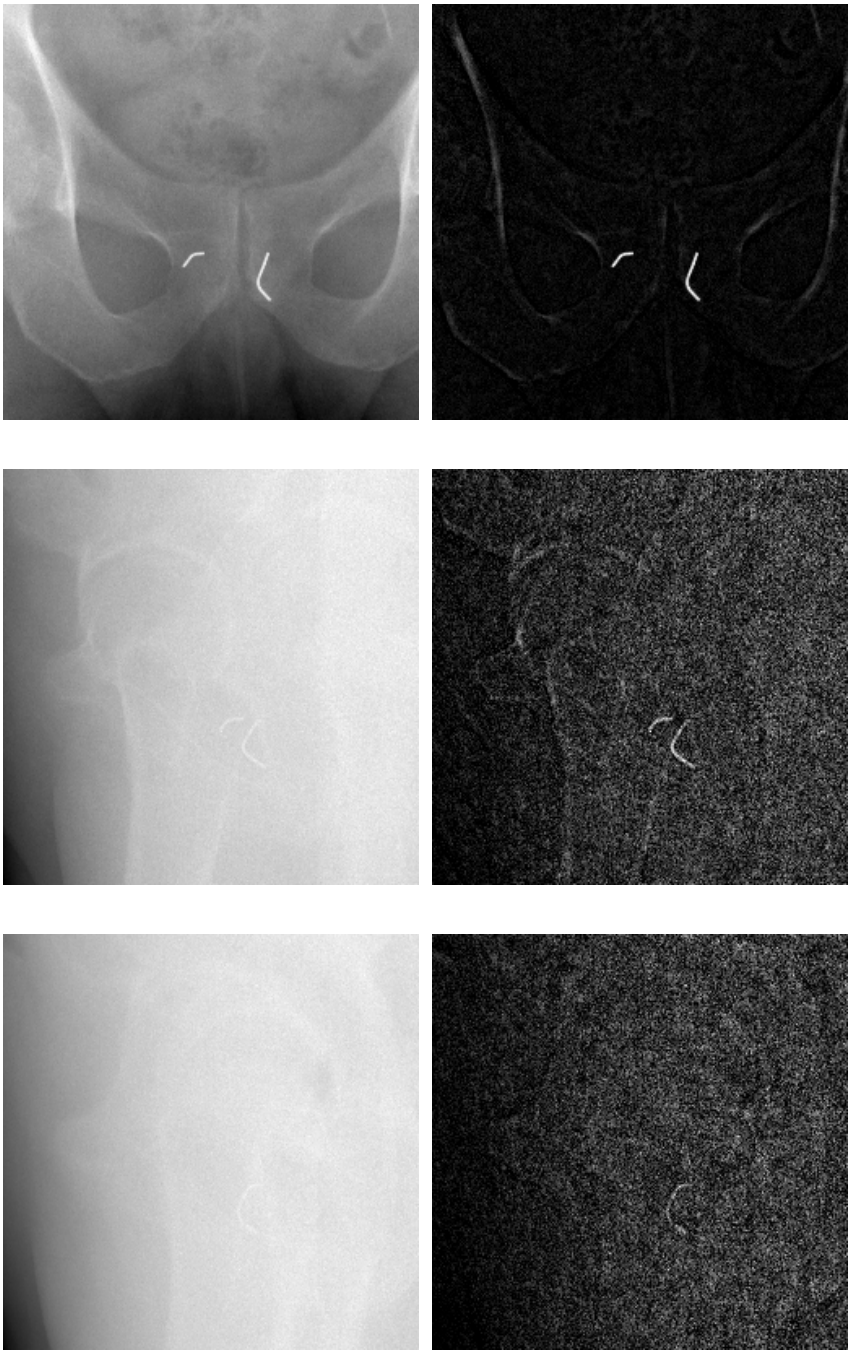
element; that is, they detect bright features which are at least five pixels long. By using eight different structuring elements, we detect linear features in any orientation.

For each top-hat transformed image, we calculate the erosion of the image with each of these linear structuring elements. The results of this erosion with three of the eight structuring elements are illustrated in Fig. 5.10 for three projection images.

ROI selection

For each projection image, we have now generated eight eroded images—one for each of the linear structuring elements. We calculate the pixel-by-pixel maximum of each of these eight images. We shall call the image thus generated the 'max-eroded' image; we have one max-eroded image for each projection image. Examples of max-eroded images are shown in Fig. 5.11a.

In the max-eroded image, the locations of linear features (regardless of orientation) have high values. Within the max-eroded image for one projection image, the higher the pixel value the 'stronger' the linear feature. We therefore find the 70 pixels which have the highest values in the max-eroded image, and call these pixels 'candidate pixels' (as they are candidates for being included in a ROI). Examples of the 70 candidate pixels are shown in Fig. 5.11b.



Original images Top-hat transformed images

Figure 5.8: Results of using the top-hat transform on three projections.

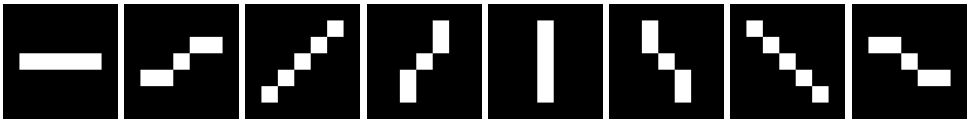


Figure 5.9: Structuring elements.

Each structuring element is of size 5×5 pixels. They are used to select linear features.

ROIs are generated to include clusters of candidate pixels. We know that markers tend to be implanted vertically, so we define the distance between candidate pixels as

$$\mathfrak{D}((u_1, v_1), (u_2, v_2)) = \sqrt{(u_2 - u_1))^2 + ((v_2 - v_1)/2)^2},$$

where u is the anterio-posterior/lateral direction (i.e. the horizontal direction, as the images are displayed in this chapter) and v is the cranio-caudal (i.e. vertical) direction. To see the effect of this definition, consider two candidate pixels separated by p pixels in the u direction, with the same v coordinate. The distance between these points is p. Compare this to two candidate pixels separated by the same amount—p pixels—in the v direction, with the same u coordinate. The distance is now $p/2$. Vertical separation is less important than horizontal separation in distinguishing between points we wish to be in different clusters (and hence in different ROIs).

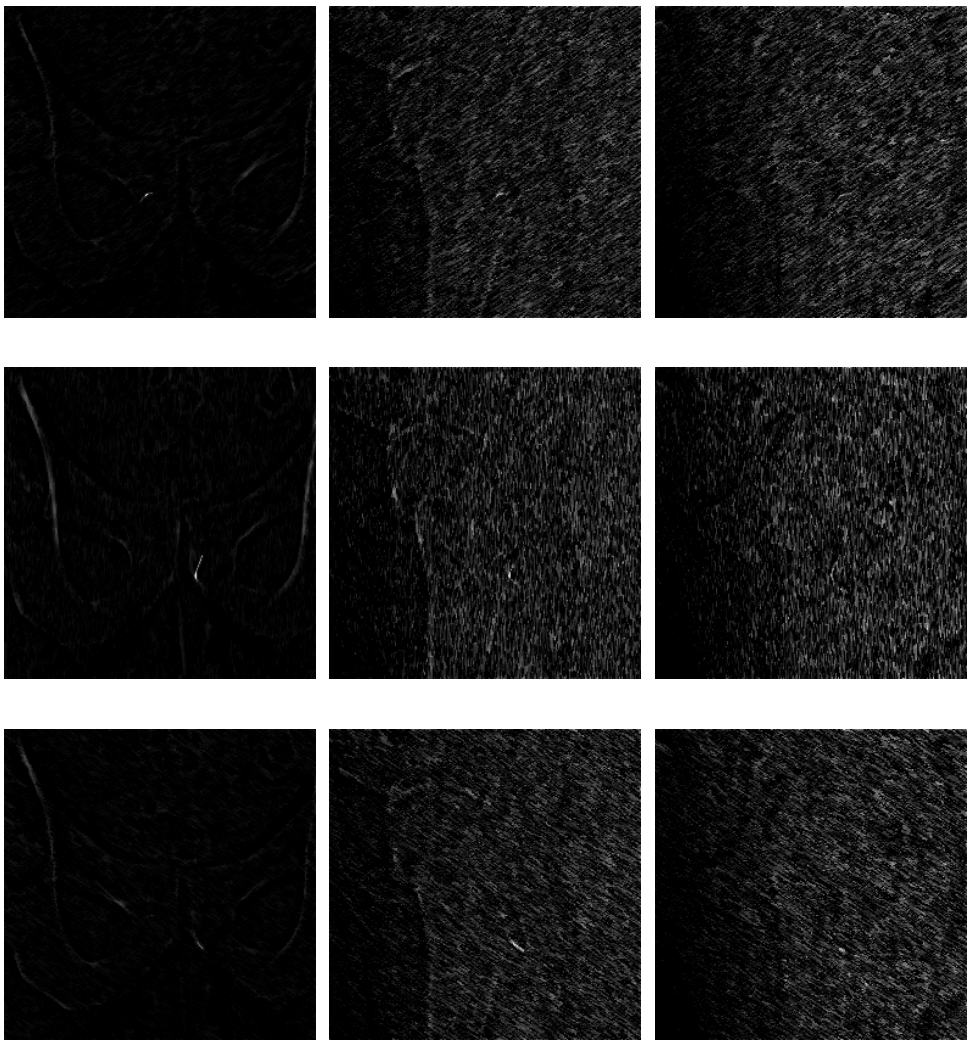
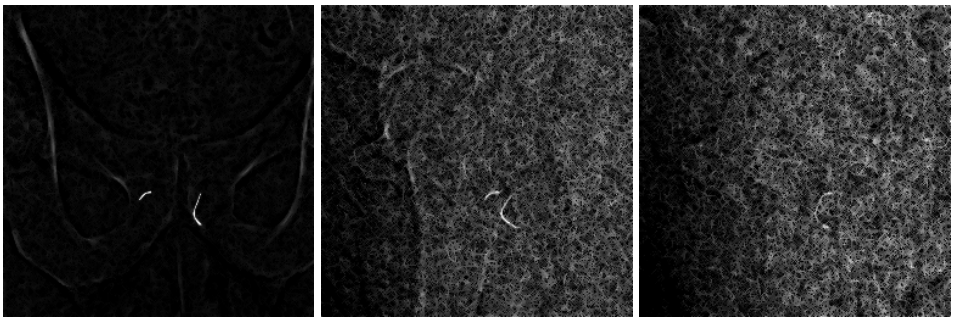


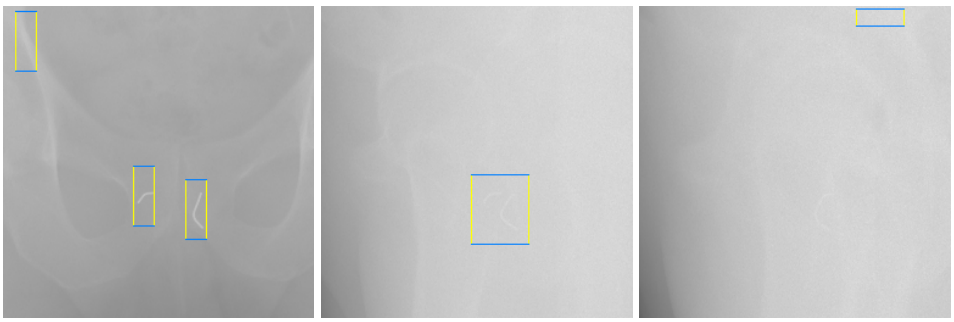
Figure 5.10: Erosion of projection images.

The erosions of three different projection images (top to bottom) with three different structuring elements (left to right) are shown. The different structuring elements detect linear features in different orientations. The projection images used are those shown in Fig. 5.8.



a: Max-eroded images.

b: The 70 candidate pixels.



c: The ROIs generated by clustering the candidate pixels.

Figure 5.11: Generation of ROIs.

The ROI generation process is shown for the same images as in Figures 5.8 and 5.10.

We generate ROIs using the following algorithm:

1. Create clusters of candidate pixels such that

$$\mathfrak{D}(a,b) < 8 \implies a \text{ and } b \text{ are in the same cluster;}$$

$$a \text{ and } b \text{ are in different clusters} \implies \mathfrak{D}(a,b) \geq 8$$

   To generate a cluster:

   (a) Pick a candidate pixel not already in a cluster and create a cluster consisting of this pixel.

   (b) Find all the candidate pixels within a distance of 8 of any of the pixels in the cluster, and include these in the cluster. Repeat this step until no more pixels are added.

   Repeat the two steps above until all candidate points are included in a cluster.

2. - If there are one or more clusters containing eight or more pixels, discard any clusters with seven or fewer pixels and retain those with eight or more.
   - If there is no cluster containing eight or more points, retain the cluster (or clusters) with the most points and discard the others.

3. For cluster c, let $u_l^c = \min\{u : (u,v) \in c\}$ (the u-coordinate of the left-most point or points in c). Similarly, let $u_r^c = \max\{u : (u,v) \in c\}$ (the u-coordinate of the right-most point or points in c), $v_b^c = \min\{v : (u,v) \in c\}$ (the v-coordinate of the bottom-most point or points in c) and $v_t^c = \max\{v : (u,v) \in c\}$ (the v-coordinate of the top-most point or points in c).

4. The width of the ROIs in this image is defined as $\max_c\{\max\{2(u_r^c - u_l^c), 10\}\}$, where $\max_c$ indicates that the maximum is taken over all clusters retained after step 2.

5. The height of the ROIs in this image is $\max_c\{\max\{2(v_t^c - v_b^c), 16\}\}$.

6. The centre coordinates of the ROI for cluster c is $((u_r^c + u_l^c)/2, (v_t^c + v_b^c)/2)$ (with both coordinates rounded down to the nearest integer).

7. If an ROI extends beyond the edge of an image, the centre coordinates are moved away from the edge so that the ROI of the specified size is entirely contained within the image.

This results in one or more ROIs in each image. Within each image, the ROIs are the same size. Examples of the ROIs generated are shown in Fig. 5.11c.

Ideally, we would have two ROIs in each image, each containing one marker. This is often the case. In Fig. 5.11, we have chosen to illustrate three images where this has not happened. In the image on the left of this figure, we have three ROIs. Both markers have been correctly identified, along with a false-positive bony edge. In the middle image, the candidate points for both markers are located in the same cluster, and hence the same ROI. In the image on the right, the markers have not been identified and do not appear in the ROI.

The parameters for the ROI generation algorithm (the number of candidate points, minimum number of points per cluster, distance between clusters) were optimised using a subset of the 'straight marker' data set, with more emphasis on reducing false negatives (markers not included in their own ROI) than on reducing false positives (ROIs which do not contain at least one marker). When applied to the other data sets ('bendy markers' and 'hooked markers'), the algorithm results in one or more false positive ROIs in approximately half the images, and misses one or more markers in approximately 15% of the images (mainly in the lateral images, such as that on the right of Fig. 5.11, where the markers are much less clear).

Note that we are currently considering each image in isolation. In Section 5.6.2 we shall combine the results obtained in each 2D image to infer information about the 3D location of the marker. Because there is information on the 3D location in many images, the later analysis can recover from false positives and false negatives generated at this stage.

5.3 Statistical model for a marker

5.3.1 Outline

Having found the ROIs which we believe have a high probability of containing markers, we define a statistical model for the markers in these ROIs. We shall

assume that there is exactly one marker in each ROI (we shall discuss the cases where this is not true later). We use a Bayesian model, first defining the prior probability distribution of a marker, then the likelihood of the data (the pixel values in an ROI) given a marker.

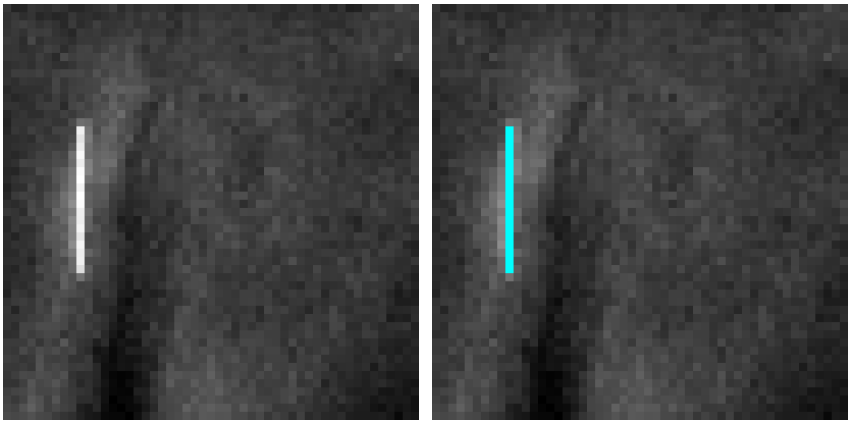
5.3.2 Prior density of marker

In the ROIs, the markers appear as approximately single-pixel-wide linear features. We have considered them as segmented features. Each segment is either one pixel high or one pixel wide or both - that is, the segment shape (h, w) (height, width) is in the set $W = \{(n_h, 1) : n_h = 1, 2, \dots\} \cup \{(1, n_w) : n_w = 1, 2, \dots)\}$. Let the pixels in the ROI be indexed by (u, v), and let $\{x_{u,v}\}$ be one if there is a marker projected at pixel (u, v) and zero otherwise. Let m be the set $\{(u, v) : x_{u,v} = 1\}$, i.e. the set of pixels where there is a marker projected. To define a possible m, we choose a starting segment and grow the marker upwards segment-by-segment. We shall illustrate this with some examples.

The first example is illustrated in Fig. 5.12. The left-hand image is part of a projection. The image on the right shows a realisation of the marker which fits these data well, with the pixels in the set m coloured cyan. The marker is a single segment of shape $(18, 1)$.

The second and third examples are illustrated in Fig. 5.13. Here, the realisations have several segments, each of which is of shape $(n_h, 1)$ (i.e. tall segments with width $= 1$). Each segment touches its neighbours at the corners. In the top example, the corners which touch neighbouring segments are diagonally opposite each other in every segment. In the bottom example, the marker 'changes direction', so that there is one segment (the long segment of shape $(12, 1)$) which has a neighbour at the bottom left and the top left.

The fourth and fifth examples are illustrated in Fig. 5.14. In these examples, parts of the markers are closer to horizontal and we need to use segments of shape $(1, n_w)$ (i.e. wide segments with height $= 1$). As in Fig. 5.13, the top example shows a realisation where all the segments touch at diagonally opposite corners, whereas in the bottom example there is a segment with neighbours at top left and bottom left.

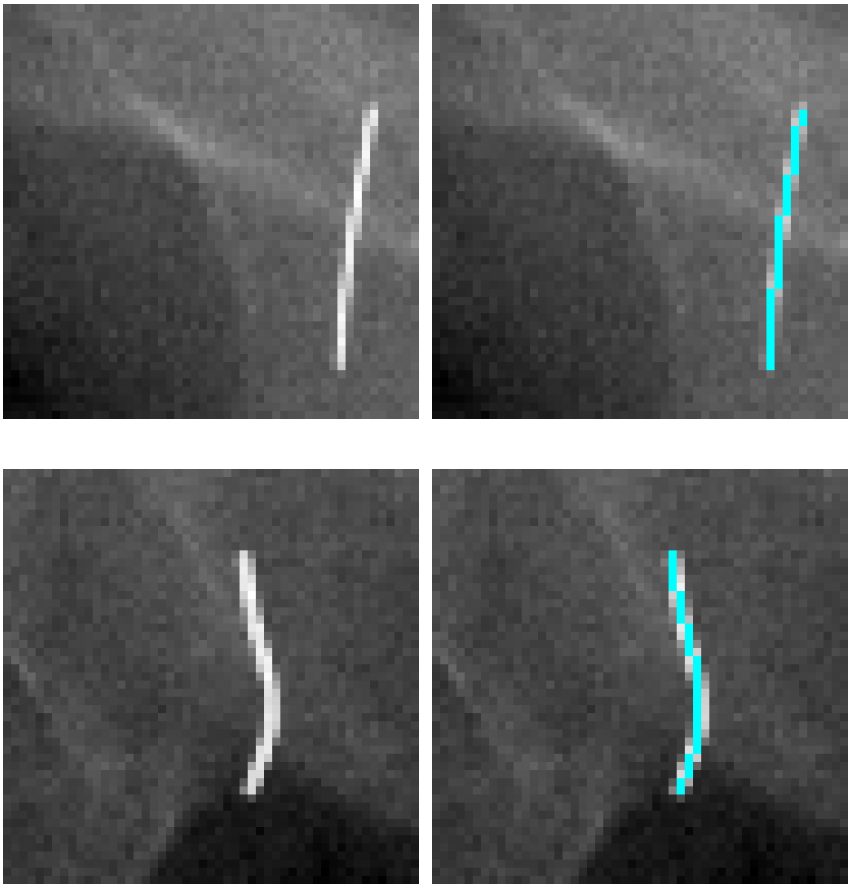


Original image. Marker pixels shown in cyan.

Figure 5.12: Illustration of a single-segment marker.

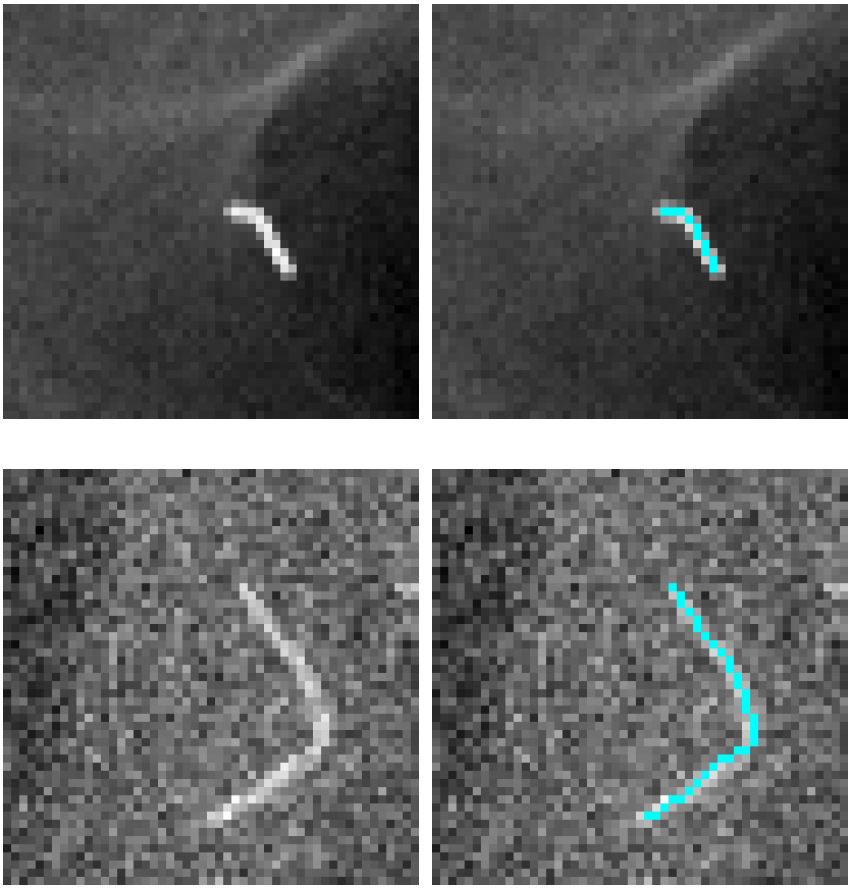
So far, these figures have shown markers which fit data. Each of these markers is deemed to be 'legal'. Fig. 5.15 shows an example of a marker which would be deemed 'illegal'. We do not allow a short, wide segment to have neighbouring segments touching at non-diagonal corners. Also illustrated are alternative legal markers, where the segment is either short and wide with diagonal neighbours, or the segment with non-diagonal neighbours is not wide.

Fig. 5.16 shows another illegal marker. The markers are implanted in such a way that they are (usually) approximately vertical, and we therefore do not allow markers where one row of the ROI contains pixels in more than one segment—each segment is clearly above or below each other segment. Note that this is different from the situation in the right-left direction, where we allow the marker to change direction, as illustrated in the right-hand image in this figure.



Original image. Marker pixels shown in cyan.

Figure 5.13: Illustration of markers consisting of long, thin segments.



Original image. Marker pixels shown in cyan.

Figure 5.14: Illustration of markers including short, wide segments.

The markers include short, wide segments and segments of shape $(1, 1)$, as well as long, thin segments.

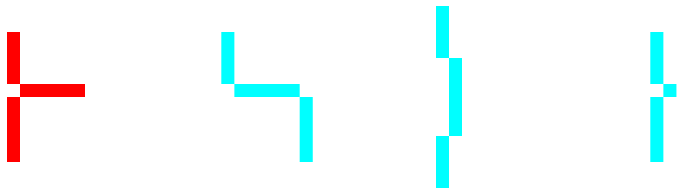


Figure 5.15: Illustration of an illegal marker.

The left-hand marker is illegal—the short, wide middle segment has neighbours at the top left and bottom left. Each of the other three markers illustrated is legal.

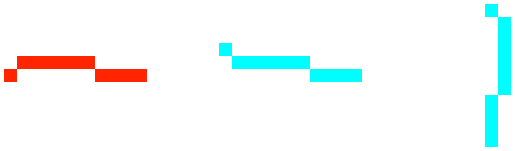


Figure 5.16: Illustration of an illegal marker.

The left-hand marker is illegal—there are two segments which contain pixels in the same row. Both the other markers illustrated are legal.

Having given these examples of legal and illegal markers, we shall now give a precise description of how to build legal markers segment-by-segment.

1. Choose a set of pixels s_1 in the ROI to be the first segment, with segment shape (h, w) from the set W as defined above, such that all the pixels in the segment are within the ROI. This will form the bottom segment of the marker. If there are at least six pixels in the first segment, this is a legal marker and it is not necessary (although possible) to add more segments.

2. Suppose another segment is to be added. Let i be the number of segments of which the marker currently consists, and hence the index of the current top segment (so $i = 1$ on the first pass). Let l_i and r_i be the left and right-most u-indices of pixels in this segment: that is, $l_i = \min\{u : (u, v) \in s_i\}$ and $r_i = \max\{u : (u, v) \in s_i\}$. Similarly, let t_i and b_i be the v-indices of the top and bottom points: $t_i = \max\{v : (u, v) \in s_i\}$ and $b_i = \min\{v : (u, v) \in s_i\}$.

3. To grow the marker from segment s_i we add the next segment s_{i+1} above segment s_i in the following way:

   (a) Choose a shape $(h, w) \in W$.

   (b) Choose a direction: right or left. Let $\Delta_i = 1$ if the direction chosen is right or -1 if the direction chosen is left. (Note that Δ_i describes the direction between the $(i-1)$th and ith segments.) If $i \leq 2$, or if segment $i-1$ has width $= 1$, we are free to choose $\Delta_i = 1$ or $\Delta_i = -1$. If segment $i-1$ has width > 1 and $i \geq 3$, we must set $\Delta_i = \Delta_{i-1}$. This enforces the rule that the transition between segments $i-1$ and i is the same direction as the transition between segments $i-2$ and $i-1$ if segment $i-1$ has width > 1.

   (c) If the new segment is to be to the right ($\Delta_i = 1$), the new segment's position is such that $(l_{i+1}, b_{i+1}) = (r_i + 1, t_i + 1)$. That is, the bottom left-most point in the new segment is one pixel higher and one pixel to the right of the top right-most point in the previous segment. If this results in pixels in the marker being outside the ROI, it it necessary to choose a different shape or direction (or to not add another segment if there are already six pixels in the marker).

(d) If the new segment is to be to the left ($\Delta_i = -1$), the new segment's position is such that $(r_{i+1}, b_{i+1}) = (l_i - 1, t_i + 1)$. That is, the bottom right-most point in the new segment is one pixel higher and one pixel to the left of the top left-most point in the previous segment. If this results in pixels in the marker being outside the ROI, it it necessary to choose a different shape or direction (or to not add another segment if there are already six pixels in the marker).

4. If the marker contains at least six pixels, it is legal and it is not necessary to add more segments.

5. Whether or not the marker contains at least six pixels, further segments may be added by repeating steps 2 and 3 one or more times. On conclusion, let the number of segments in the marker be N_s; the marker $m = \bigcup_{i=1}^{N_s} s_i$.

Within an ROI of a fixed size, there are a finite number of possible markers of this kind, say N_k. Let these possible markers be denoted m_k for $k = 1 \ldots N_k$, and let M be a random sample from $\{m_k\}$. We now describe a prior probability distribution for M, based on the physical properties of the markers. We wish to penalise shortness, wiggliness and curviness. For a realisation of the marker $M = m$, we define these as follows:

- **Shortness, $a(m)$.** The markers are inserted into the patient so that they are approximately vertical. In the projections, they are therefore not excessively short. We set $a = \max\{|m| - 16, 0\}$, where $|\cdot|$ denotes the size of the set m. Thus a is the number of pixels by which the size of the marker is less than 16.

- **Wiggliness, $b(m)$.** The markers have a certain physical stiffness. They may curve and change direction (as in Fig. 5.14), but do not generally bend in one direction and then back the other way. Recall that $\Delta_i \in \{1, -1\}$ is the direction of the transition between segment $i-1$ and segment i (representing 'right' or 'left'). We set $b(m)$ to be one less than the number of times the marker changes direction (or zero if the marker does not change direction); that is

$$b(m) = \max\{\text{changes} - 1, 0\}$$

where

$$\text{changes} = |\{i \text{ s.t. } \Delta_i \neq \Delta_{i-1} \; ; i = 2 \ldots N_s\}|.$$

- **Curviness, $c(m)$.** Again, because of the stiffness of the markers, they are generally reasonably straight. We wish to penalise markers where the shapes of adjacent segments are markedly different. We do not wish to penalise markers where the difference in the height or width of adjacent segments is one pixel (which can occur in a perfectly straight marker because of the discretisation inherent in our model). We define the curviness in such a way that the marker can have segments which are one pixel longer or wider than their neighbours without penalty, but any greater difference than this is penalised in proportion to the number of pixels in excess of one that the heights or widths differ. A smoothly curving marker is penalised less than one in which there is a sharp angle. To do this, define

$$\Lambda(i) = \begin{cases} h_i - 1 & \text{if } h_i > 1 \\ -(w_i - 1) & \text{if } w_i > 1 \\ 0 & \text{otherwise.} \end{cases}$$

  We are interested in the differences between shapes; the definition of $\Lambda(i)$ allows us to deal smoothly with the transition from segments with height $>$ 1 through segments that have height $=$ width $=$ 1 to segments with width $>$ 1. We then set

$$c(m) = \sum_{i=2}^{N_S} \max\{|\Lambda(i) - \Lambda(i-1) - 1|, 0\}.$$

With the characteristics a, b and c defined above, we combine them in a prior probability of the form

$$\mathbb{P}(M = m) \propto \exp\{-(\beta_a a(m) + \beta_b b(m) + \beta_c c(m))\} = f(m) \tag{5.1}$$

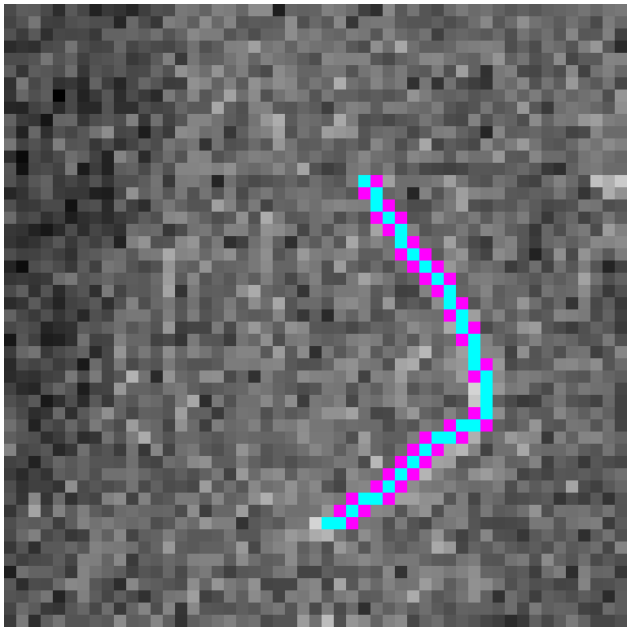


Figure 5.17: Illustration of the 'half' pixels.

The 'half' pixels are shown in magenta, with the marker pixels (as shown in Fig. 5.14) in cyan.

where the β values are parameters to tune the amount of penalisation. The values of β_a, β_b and β_c were selected empirically to be 0.5, 5.0 and 0.2 respectively.

5.3.3 Likelihood of data given marker

We shall now define the likelihood of the observed pixel values in an ROI, given the location of exactly one marker in that ROI. Let the measured pixel value at pixel (u, v) be $y_{u,v}$. As elsewhere in this thesis, we assume that the $y_{u,v}$ are independent given $M = m$, with $y_{u,v}|M = m$ normally distributed. In this section, the choice of parameters for the normal likelihood will be described.

We consider three different categories of pixels: those in the marker, those which we shall call 'background', and a third category which we shall call 'half' pixels. The segments meet corner to corner; the 'half' pixels are defined as the other pixels which share a corner with two segments. This is illustrated in Fig. 5.17.

To properly define these categories, recall that the set m was defined to be the pixels (u, v) for which $x_{u,v} = 1$, where x indicates whether there is a marker

projected at pixel (u, v). We define the set of 'half' pixels, h, to be those pixels (u, v) where at least one of the following four statements is true:

$$(u, v+1) \in m \quad \text{and} \quad (u+1, v) \in m$$

$$(u, v-1) \in m \quad \text{and} \quad (u+1, v) \in m$$

$$(u, v+1) \in m \quad \text{and} \quad (u-1, v) \in m$$

$$(u, v-1) \in m \quad \text{and} \quad (u-1, v) \in m.$$

Note that, by our definition of m, such points $(u, v) \notin m$. We define the background pixels to be $b = \{(u, v) : (u, v) \notin m \cup h\}$. Note that every pixel in the ROI is either in m, h or b, and that once the pixels contained in m have been defined, the categorisation of all other pixels in h or b follows.

The local background mean μ_b was estimated by the median pixel value for the pixels at the edge of each ROI. We define the mean for the 'marker' pixels as $\mu_m = 2500 + 0.775\mu_b$. This formula was chosen to approximate the observed values. Recall that the image data have already been transformed to be on the log scale, so that (in the absence of scatter) we would expect to see an approximately constant difference between the background and marker mean values. The presence of scatter causes the difference to be less in the lateral projections (where there is a greater thickness of tissue), which is where the background mean is higher. Rewriting the expression relating μ_b and μ_m as $\mu_m - \mu_b = 2500 - 0.225\mu_b$ makes clear the relationship between the difference between the marker and background means and the background mean.

We define the mean value for the 'half' pixels as the average of the marker and background mean values. Within each ROI, the values of the 'marker', 'half' and 'background' pixels were assumed to have a common variance, σ^2. This was estimated by the variance of the pixel values at the edge of the ROI. In doing so, we assume that the pixels at the edge of the ROI do not represent a marker. Their variance gives an indication of how much the background varies, whether due to the presence of anatomical structure or random noise.

As previously stated, we assume a normal likelihood. Thus,

$$y_{u,v}|M \sim \begin{cases} \mathcal{N}(\mu_b, \sigma^2) & \text{if } (u,v) \in b \\ \mathcal{N}(\mu_h, \sigma^2) & \text{if } (u,v) \in h \\ \mathcal{N}(\mu_m, \sigma^2) & \text{if } (u,v) \in m, \end{cases} \tag{5.2}$$

where $\mu_m = 2500 + 0.775\mu_b$ and $\mu_h = (\mu_b + \mu_m)/2$.

5.3.4 Posterior

In Section 5.3.2 we have defined a set of possible markers within an ROI of a certain size, $\{m_k : k = 1, \ldots, N_k\}$, and defined the prior probability distribution for these markers, $f(m)$ (up to a constant of proportionality). In Section 5.3.3, we have also defined the likelihood of the observed pixel values in an ROI given a marker. The posterior probability distribution of the marker given the data is proportional to the product of the prior and the likelihood. We shall denote the posterior probability of m, $\mathbb{P}(M = m|Y = y)$, by $\pi(m|y)$.

5.4 Analysis of posterior density

In order to provide useful information about the markers, we need to estimate the location of each the four end points (as a proxy measurement of the location of the prostate gland). In this section, we shall describe how we have done this using an MCMC sampler to find the probable markers.

5.4.1 MCMC sampler

We have used a Metropolis-Hastings sampler to explore the posterior density of the markers within each of the ROIs. In order to construct a Markov chain which moves around the space of possible markers, we have devised a number of moves to perturb the marker parameterisation in Section 5.3, giving a new marker.

The moves are illustrated in Figures 5.18 to 5.22. They are defined as:

- **Move marker:** Move the whole marker by one pixel up, down, left or right (Fig. 5.18).
- **Switch direction:** Change the direction of the shift between two segments. The directions of the shift between the subsequent segments stays the same (Fig. 5.19).
- **Flip:** Select a set of two or more consecutive directions Δ_i and multiply each one by -1 (thus changing the directions in a part of the marker) (Fig. 5.20).
- **Grow:** Add a segment of shape $(1 \times 1), (1 \times 2), (1 \times 3), (2 \times 1)$ or (3×1) to either end of the marker (Fig. 5.21).
- **Shrink:** If the first or last segment has one of the shapes listed in 'grow segment', remove it (Fig. 5.21).
- **Sharpen corner:** If there is a (1×1) segment where the directions between the preceding segment and this segment, and between this segment and the succeeding segment are the same, remove the (1×1) segment by increasing the height (or width) of the preceding segment and the width (or height) of the succeeding segment (Fig. 5.22).
- **Smooth corner:** If there is a segment which has height > 1 followed by a segment with width > 1, reduce the height of the former and the width of the latter, and add a new (1×1) segment to connect them. Similarly, if there is a segment with width > 1 followed by a segment with height > 1, reduce the length of these segments and add a (1×1) segment to connect them (Fig. 5.22).
- **Move boundary:** Move the boundary between two segments by one pixel (making one segment longer and the other shorter). This includes the boundary at the end of the marker, so the final segment may grow or shrink. In Fig. 5.23, two possibilities are illustrated—one moving a boundary vertically, and one horizontally.

- **Insert small:** Create a segment of shape (1×1) to 'pop out' of a segment of length ≥ 3. Do this by changing the categorisation of a pixel not at the end of the segment to no longer being in m (it will in fact now be in h), and changing the categorisation of a pixel either to the right or left to be in m. We thus create three segments from the original long segment (Fig. 5.24).

- **Remove small:** If there is a segment of shape (1×1) sticking out of the marker, remove it by changing the categorisation of this pixel to be not in m and changing the categorisation of the pixel to its right or left to be in m, thus creating one long segment from the (1×1) segment and the preceding and succeeding segments (Fig. 5.24).

- **Pop end out:** Create a (1×1) segment at the end of the marker by removing the final pixel from m, and adding the pixel representing the new segment. The new segment may be to the left or right, or (at the top of the marker) higher, or (at the bottom of the marker) lower (Fig. 5.25).

- **Pop end in:** If the first or final segment is (1×1), remove it and increase the height or width of the neighbouring segment by one (Fig. 5.25).

- **Insert wiggle:** If there is a segment with height > 1 followed by a segment with height $= 1$, create a (1×1) segment between them. To do this, reduce the height of the first segment by one, create the (1×1) segment with the direction opposite to the direction between the original two segments, and increase the width of the final segment (Fig. 5.26).

- **Remove wiggle:** If there is a (1×1) segment where the preceding segment has width $= 1$, the succeeding segment has width > 1, and the directions between the preceding segment and this segment, and between this segment and the succeeding segment differ, remove the (1×1) segment. To do this, increase the height of the preceding segment and reduce the width of the succeeding segment (Fig. 5.26).

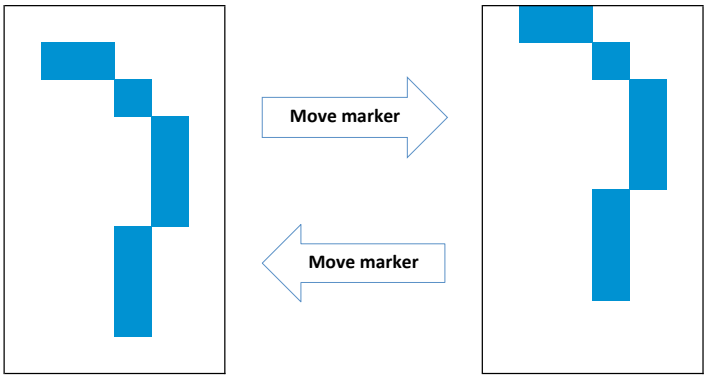


Figure 5.18: Move marker.

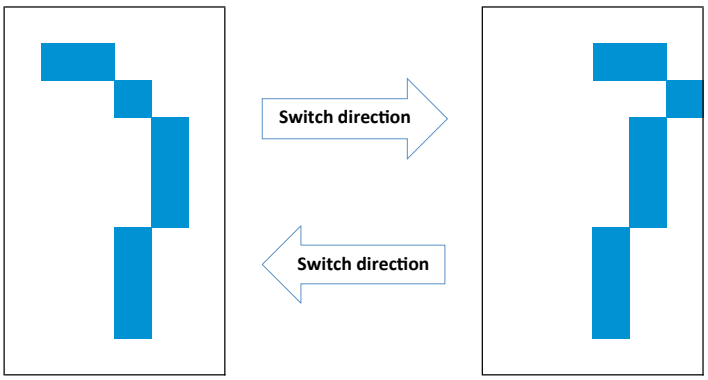


Figure 5.19: Switch direction.

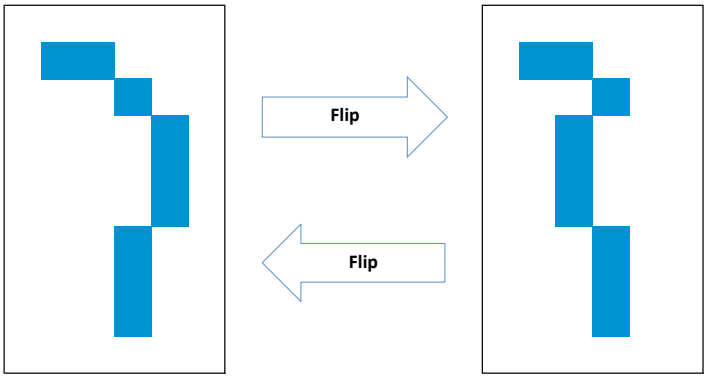


Figure 5.20: Flip.

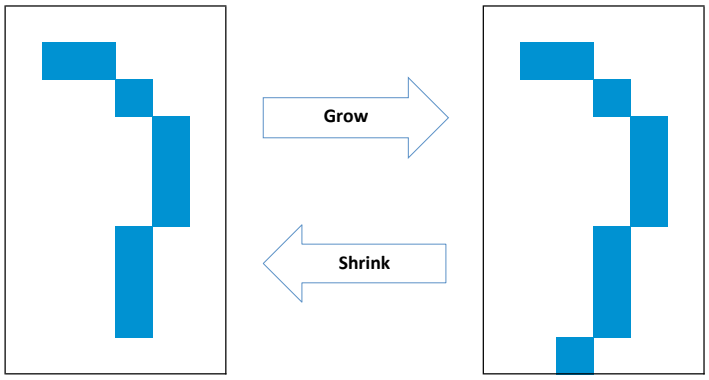


Figure 5.21: Grow / shrink.

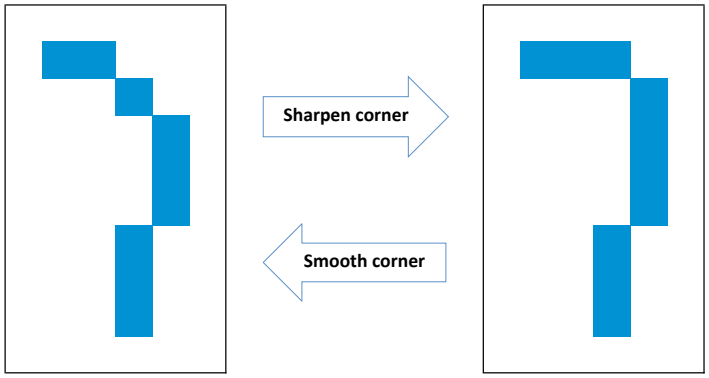


Figure 5.22: Sharpen / smooth corner.

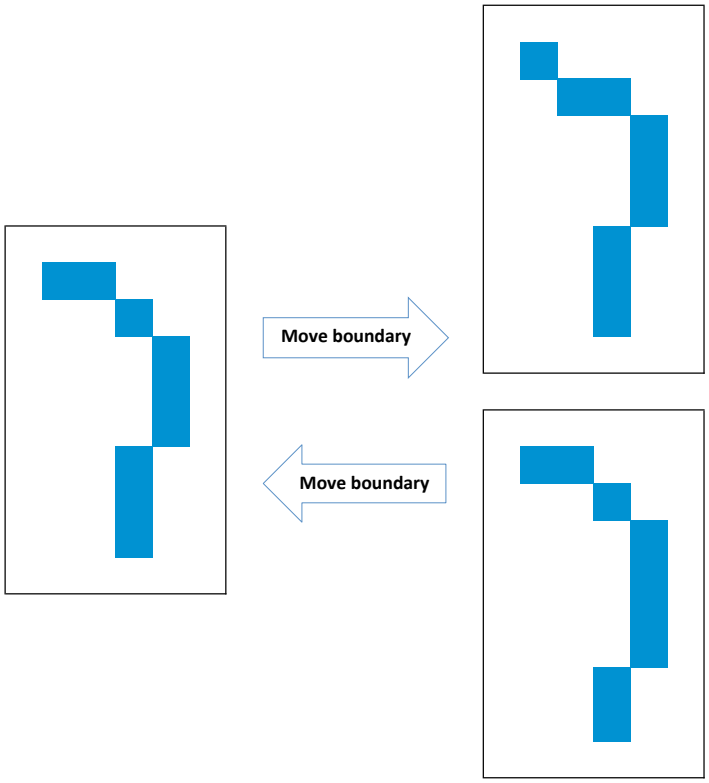


Figure 5.23: Move boundary.

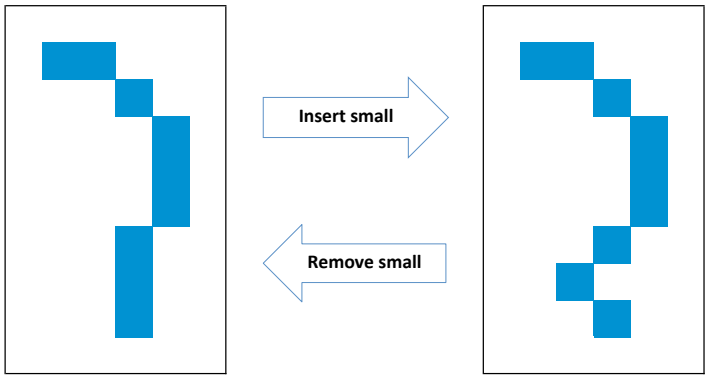


Figure 5.24: Insert small / remove small.

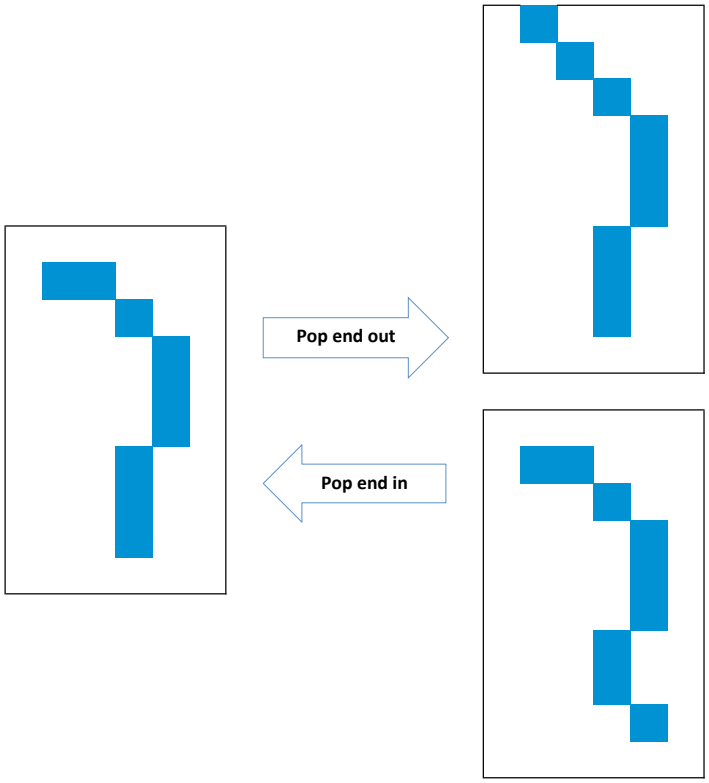


Figure 5.25: Pop end in / pop end out.

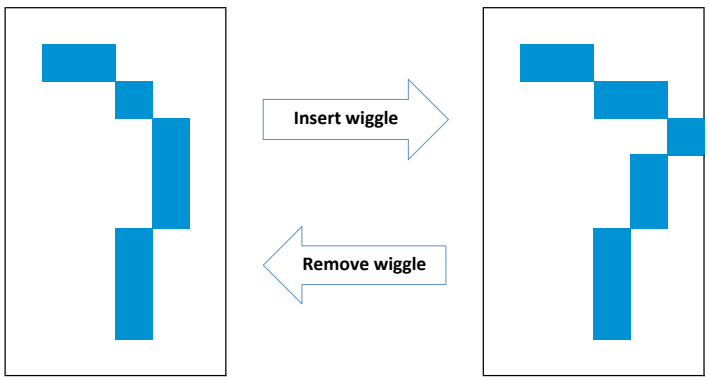


Figure 5.26: Insert wiggle / remove wiggle.

A smaller set of moves would create a Markov chain that is irreducible. However, without moves such as *remove wiggle*, the sampler was prone to getting trapped for long periods in local modes—to get to the state resulting from a *remove wiggle* move, we would need to go via other states where the likelihood is much lower. The inclusion of this move therefore allows the sampler to escape from local modes more easily.

The probabilities assigned to each of the moves was 0.1, except for *smooth/sharpen corner*, *insert/remove small segment*, *pop end in/out*, and *insert/remove wiggle*, which were each assigned a probability of 0.05. In some cases, it may not be possible to carry out all the moves; for example there may not be a suitable configuration of segments to carry out a *remove wiggle* move. In this case, the probability assigned to the impossible move or moves is divided equally between the possible moves. Once a move type has been chosen, there are usually many choices of how to carry out the move; for example we need to choose the direction in which to move in *move marker* or where to insert a segment in *insert small segment*. Each of the possible options was chosen with equal probability.

In order to make the chain reversible, we must calculate the Hastings ratio at time t

$$\frac{\pi(m^\star|y)q(m^{(t)} \to m^\star)}{\pi(m^{(t)}|y)q(m^\star \to m^{(t)})},$$

where $q(a \to b)$ is the probability of proposing state b when the chain is in state a.

We first consider $q(m^{(t)} \to m^\star)$, the probability of proposing $m^\star$ from $m^{(t)}$. The moves are designed in such a way that there is a unique single move between these two states. After redistributing the probability from the moves which are impossible from state $m^{(t)}$, we can calculate $q(m^{(t)} \to m^\star)$ as the probability for the required move, divided by the number of possible ways to carry out this move.

For example, consider the transition represented in Fig. 5.24. The base probability for the move *insert small segment* is 0.05. For the marker on the left of this figure, it is not possible to carry out moves of type *smooth corner*, *remove small segment*, *pop end in* or *remove wiggle*. These impossible moves each have a base probability of 0.05, so the total base probability of 0.2 is redistributed over the remaining ten possible moves. The probability of choosing *insert small segment*

is therefore $0.05 + 0.2/10 = 0.07$. There are four different ways in which a small segment could be inserted—we could turn each of the three-pixel-high segments into three 1×1 segments by creating the new segment to the right or left of the original segment. The final calculation is therefore $q(m^{(t)} \rightarrow m^{\star}) = 0.07/4 = 0.0175$.

We now consider the probability for the reverse move, $q(m^{\star} \rightarrow m^{(t)})$. In our case, each move may be reversed by itself or another move. We therefore need to find this move, and make the same calculation as for the forward probability.

In our example, the base probability for the reverse move, *remove small segment*, is 0.05. There are two impossible moves from the proposed marker, with a total base probabily of 0.1. The probability of choosing *remove small segment* is therefore $0.05 + 0.1/12 = 0.0583$. There is only one possible way to carry out the *remove small segment* move, so our value for $q(m^{\star} \rightarrow m^{(t)})$ is 0.0583.

5.4.2 Comparison of enumeration and sampling approaches

As has already been noted, the pixel-based nature of the model used in this chapter means that there are a finite number of possible realisations of the marker M in a given ROI. In this section, we shall demonstrate that the posterior probability is concentrated on relatively few markers (in some cases, on very few). In order to achieve our aims of estimating the position of the ends and the pixels in which there is likely to be signal due to a marker, it is sufficient to identify these few markers. We shall use an MCMC sampler to find high-probability markers.

For a very small ROI (much smaller than the ROIs generated from the projection images), it is possible to enumerate $\{m_k : k = 1 \ldots N_k\}$ and calculate the posterior probability for each m exactly. In this section, we shall carry out this enumeration for markers in three ROIs designed to emulate three different situations: a clearly visible marker, a false positive (as where an ROI includes a bony edge), and a barely visible marker (as in a lateral projection).

We shall calculate the posterior probability that each pixel in the small ROIs represents the end of a marker, and then compare this with the estimates obtained by using an MCMC sampler as a means of identifying the few probable markers.

Suppose we have a probability distribution (or mass) function $f(x)$, and an event A for which we wish to know the probability. If we construct a Markov chain with $f(x)$ as its stationary distribution, the usual way to estimate the probability that A occurs is as the proportion of time (after some period of burn-in) that the chain is in a state where A occurs.

We do not use the chain in this way. Instead, we rely on the fact that the chain will tend to sample regions of relatively high probability, and use it as a means to identify high-probability states. We maintain a list of every unique marker which is either visited or proposed by the chain. Because we are dealing with a discrete set, we can find the posterior probability of these markers exactly (up to a constant), and then normalise based on the states considered. In this situation, the method we use is more accurate than the usual estimate formed by considering the amount of time the chain remains in a particular state.

Because we are not relying on sampling of the stationary distribution of the Markov chain, we do not need to discard markers as burn-in, in the traditional sense. However, there is a cost associated with storing a marker and its probability, and particularly with comparing markers with our stored list as we propose and visit potential new markers. We therefore choose to discard the first iterations, after which the sampler is more likely to be in a high-probability region than at initialisation.

'Clear' ROI

In order to make enumerating the markers possible, we used very small ROIs of size 14×4 pixels. Within an ROI of this size, there are $N_k = 10,140,826$ possible markers. We generated ROI data to simulate the three situations of a clear marker, a false positive and an unclear marker.

We shall first consider the clear marker. Fig. 5.27 shows part of a projection image showing a clear marker, and the small ROI which is intended to emulate this clear marker. We shall call this 14×4-pixel ROI the 'clear' ROI.

For this ROI, there is one realisation of the marker which is much more probable than any other—in fact the posterior probability $\pi(m|y)$ is greater than 0.9999 for

this marker. This marker and the next most probable marker, with conditional probability 1×10^{-6}, are shown in Fig. 5.28.

We shall now demonstrate how an MCMC sampler may be used to find the probable markers (in this case, the single probable marker). The sampler described in Section 5.4.1 was run several times. In each case, the first 2000 iterations were discarded, with the sampler then being run for a further N iterations for $N = 50, 100, 200, 500, 1000, 2000, 5000, 10000$. All markers which the sampler either visited or proposed (but did not accept) were recorded, along with their posterior probability.

In one series of experiments, the sampler was run once, initialised with a marker from corner to corner. In a second series of experiments, the sampler was run six times, with different initial markers. In this second series of experiments, the list of markers was comprised of all those proposed or visited in any of the six runs. Each of the series was repeated 100 times for each ROI for each run length. The six initial markers are shown in Fig. 5.29; the single-run experiments were all initialised with the first of these.

The results of these experiments for the 'clear' ROI are shown in Table 5.2. The figures given are the number of runs (out of the 100 repeats) where the markers which were visited or proposed by the sampler account for more than 99% of the probability mass (which, for this ROI, means the number of runs where the most probable marker was either visited or proposed.)

In this case, the length of the run appears to have little effect on whether the high-probability marker is found. Running the sampler six times from different start points is a reliable way of finding the probable marker, even for very short runs (after the initial 2000 iterations).

Part of a projection image containing a clear marker.

Small ROI designed to emulate the clear marker.

Figure 5.27: ROI containing a clear marker.

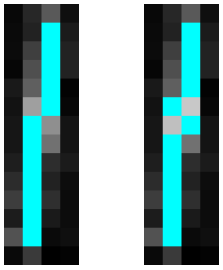


Figure 5.28: The two most probable markers for the 'clear' ROI.

	Iterations (after initial 2000)							
	50	100	200	500	1000	2000	5000	10000
Single runs with set representing > 99%	84	69	78	79	70	68	70	69
Single runs with set representing > 95%	84	69	78	79	70	68	70	69
Sets of six runs with set representing > 99%	100	100	100	100	100	100	100	100
Sets of six runs with set representing > 95%	100	100	100	100	100	100	100	100

Table 5.2: Results from MCMC runs on the 'clear' ROI.

The numbers in the table give the number of individual runs or sets of runs where the sampler found a set of markers representing the given percentage of the probability mass.

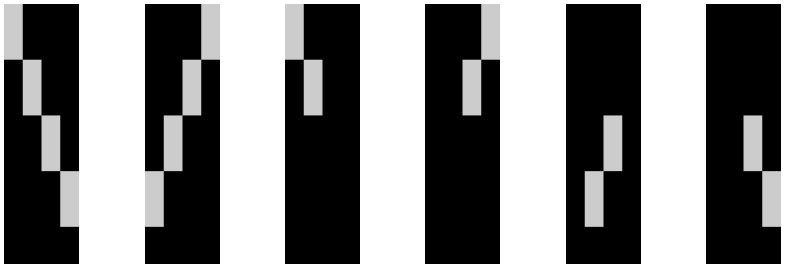


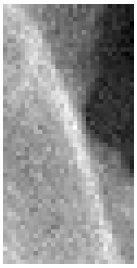
Figure 5.29: The markers used to initialise the MCMC samplers.

'False' ROI

We shall now consider the false-positive ROI. Fig. 5.30 shows part of a projection image showing the bony edge of the pelvic girdle. Such edges may cause a false-positive ROI in the morphological analysis. Also shown is the small ROI which is intended to emulate this bony edge. We shall call this 14×4-pixel ROI the 'false' ROI.

For this ROI, there are two markers with high posterior probabilities (0.49 and 0.47), while the third most probable marker has probability 0.009. These three markers are shown in Fig. 5.31. The most probable nine markers account for a total probability of more than 0.99.

To investigate the ability of the MCMC sampler to find the high-probability markers, the same series of experiments as described above were run on the 'false' ROI. The results for the 'false' ROI are shown in Table 5.3; in this case the sampler was able to find sets accounting for 95% of the probability mass. The number of runs in which the sampler finds a set of markers representing a particular proportion of the total posterior probability increases as the sampler length increases. It is also clear that using multiple runs from different starting points is again helpful in finding high-probability markers.



Part of a projection image containing a bony edge.

Small ROI designed to emulate the bony edge.

Figure 5.30: ROI containing a false-positive marker.

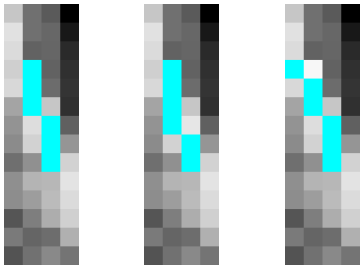


Figure 5.31: The three most probable markers for the 'false' ROI.

	Iterations (after initial 2000)							
	50	100	200	500	1000	2000	5000	10000
Single runs with set representing > 99%	0	0	0	0	1	1	7	15
Single runs with set representing > 95%	55	54	63	66	68	71	84	89
Sets of six runs with set representing > 99%	37	69	90	95	94	92	96	96
Sets of six runs with set representing > 95%	100	100	100	100	100	100	100	100

Table 5.3: Results from MCMC runs on the 'false' ROI.

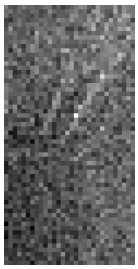
The numbers in the table give the number of individual runs or sets of runs where the sampler found a set of markers representing the given percentage of the probability mass.

'Unclear' ROI

Fig. 5.32 shows part of a projection image showing a marker in a lateral projection. The marker is very unclear. Also shown is the small ROI which is intended to emulate this situation. We shall call this 14×4-pixel ROI the 'unclear' ROI.

For this ROI, the probability distribution is much more dispersed. The most probable marker has probability 0.015. The most probable 290 markers account for a total probability of 0.5; the first 29, 774 account for a total probability of 0.99.

When the same series of MCMC experiments were run on this ROI, none of the single runs or multiple runs found a set of markers accounting for more than 90% of the probability mass. The results are shown for 85%, 80% and 50% in Table 5.4. However, we are interested primarily in the probability that each pixel contains the end of a marker (this will be discussed in detail later). We have used the list of markers generated by six runs of 2000 iterations to create an estimate of this probability for each pixel. This is shown in Fig. 5.34. In the series of 100 replications of this sampling scheme, none of the samplers found a set of markers which accounted for more than 80% of the posterior probability. However, the pixel-by-pixel estimates of the probability that there is an end at that pixel are good.



Part of a projection image containing an unclear marker.

Small ROI designed to emulate the unclear marker.

Figure 5.32: ROI containing an unclear marker.

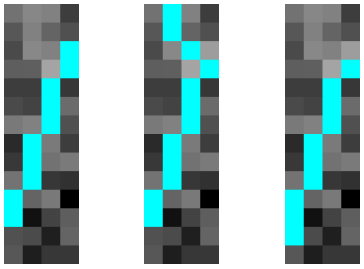


Figure 5.33: Three highly-probable markers for the 'unclear' ROI.

The markers shown are the first, tenth, and hundredth most probable.

	Iterations (after initial 2000)							
	50	100	200	500	1000	2000	5000	10000
Single runs with set representing > 50%	0	0	0	0	0	0	6	100
Sets of six runs with set representing > 85%	0	0	0	0	0	0	0	100
Sets of six runs with set representing > 80%	0	0	0	0	0	0	11	100
Sets of six runs with set representing > 50%	0	0	0	0	60	100	100	100

Table 5.4: Results from MCMC runs on the 'unclear' ROI.

The numbers in the table give the number of individual runs or sets of runs where the sampler found a set of markers representing the given percentage of the probability mass.

True values from enumeration. Estimates from MCMC sampler.

Figure 5.34: End probabilities calculated by enumeration and sampling.

For each pixel, the greyscale indicates the probability that this pixel contains a marker end for the 'unclear' ROI. White corresponds to a probability of 1, and black to a probability less than 1×10^{-10}, with the intervening grey values on a log scale.

Discussion

These results demonstrate that, although the probability of obtaining an accurate estimate of the ROI probability increases with the number of iterations, it is also important to run multiple samplers and collate the results from these multiple runs. Each run tends to get stuck for some time in local modes, and running several samplers allows us to find the significant modes with fewer overall iterations than running one sampler for long enough to escape them.

Note that we are not using the proportion of time that the sampler spends in any state to estimate the probability of that state. This means that we are not concerned by the length of time for which a chain is stuck in a particular mode—we are not relying on averaging the sample to determine the probability of each mode. This applies both within the results from one run of the sampler, and for the results of several runs—averaging over replications of poorly-mixing chains would not be a good solution, since we would still not know the relative probability of each mode.

In some cases (particularly the case of a clear marker), we obtain very accurate results. In others, we have shown that despite finding markers representing only 80% of the probability mass, we have been able to generate a reasonably accurate pixel-by-pixel map of the probability that each pixel contains a marker end.

These results have also informed our choice of sampler chain length for the real data. In Section 5.6.3, we shall show results using a sampler which is run six times on each ROI, with the six starting points being from corner to corner or corner to centre (in a similar fashion to Fig. 5.29). We record all the markers visited or proposed in the 5000 iterations after discarding the first 2000.

5.5 Creating the 2D probability map

5.5.1 Overview

In Section 5.4, we described an MCMC sampler, and how we can use this sampler to find markers with high posterior probability. We are interested primarily in the positions of the end points of the markers.

Later in this chapter (Section 5.6), we shall combine the information from many projection images to create estimates of the 3D location of the marker ends. In order to do this in a meaningful way, we shall require a non-zero probability (representing the posterior probability that this pixel contains the projection of a marker end) assigned to each pixel in the projection images (or, at least, non-zero values in the areas of the images which represent the projection of a finite volume of the patient in which we expect to find the markers).

Taken alone, the ROI selection process and sampling approach would result in some pixels (e.g. all those not included in an ROI) having zero probability. However, the ROI selection and sampling are not infallible, so it makes sense to include the possibility of error in either of these processes in our model for the 2D projection data.

In this section, we shall describe how we assign a non-zero posterior probability to each pixel in the image, representing the probability that this pixel contains the end of a marker, given the data.

5.5.2 Pixels in ROIs

For each ROI, we shall now form an estimate of the posterior probability that there is a marker end at each pixel within the ROI. This will be based on the list of probable markers generated by the MCMC sampler.

We define

$$p^{\star}_{\mathcal{A}}(u,v) = \mathbb{P}(\text{marker end at pixel } (u,v) | \text{marker in ROI } \mathcal{A}, y). \tag{5.3}$$

We can estimate this from our sampler output. Let $f(m|y)$ be the unnormalised posterior probability of marker x. Let $\mathbb{I}(A)$ be the indicator of the event A (that is, $\mathbb{I}(A) = 1$ if A occurs, and 0 otherwise). Our estimate of $p^{\star}_{\mathcal{A}}(u,v)$ is then

$$\hat{p}^{\star}_{\mathcal{A}}(u,v) = c \sum_{k=1}^{N_m} f(m_k|y)\mathbb{I}(m_k \text{ has an end at pixel } (u,v)), \tag{5.4}$$

with the sum being over the N_m markers visited or proposed by the MCMC sampler, and the constant c calculated such that $\sum_{(u,v)} \hat{p}^{\star}_{\mathcal{A}}(u,v) = 1$.

There are a number of reasons why this may not be an accurate estimate. Firstly, we have assumed that there is one marker in the ROI. We have seen in Fig. 5.11 that this is not always the case - there may be zero, one or two markers in the ROI. The accuracy of our estimate also depends on the sampler visiting or proposing realisations of markers which represent the truth well, with all high-probability realisations included in the summation in Equation 5.4.

We therefore introduce the quantity ε_1 to represent the probability that the sampler has for some reason failed to identify a projected marker end (either due to the failure of our model by not allowing two markers in an ROI, or because it has never reached high-probability realisations). We formulate another estimate of $p^{\star}_{\mathcal{A}}(u,v)$,

$$\tilde{p}^{\star}_{\mathcal{A}}(u,v) = \frac{\varepsilon_1}{N_{\text{ROI}}} + (1-\varepsilon_1)\hat{p}^{\star}_{\mathcal{A}}(u,v), \tag{5.5}$$

where N_{ROI} is the number of pixels in each ROI.

As discussed in Section 5.5.1, we require the estimated posterior probability assigned to each pixel to be non-zero. When these are combined in 3D, this will allow the marker ends to be located at points which would otherwise be impossible. In practice, the exact size of ε_1 is not important, as long as it is small when compared to the estimated posterior probability for a pixel which had a high probability assigned by the sampler. The same applies to ε_2, which we shall describe next. For most points in 3D space, we shall find that the probability of there being a marker at that point is the product of many ε_1s or ε_2s: effectively zero.

5.5.3 Pixels not in an ROI

In Section 5.5.2, we have calculated an estimate of the probability that any pixel in an ROI contains the end of a marker. We now turn our attention to the pixels which are not included in an ROI.

We assume that each of the two markers appears in an ROI with probability $(1 - \varepsilon_2)$, and that the probabilities of correctly locating an ROI containing each of the two markers are independent. For any pixel not in an ROI, we define the probability that the end of a marker is in that pixel as

$$p_{\text{outside}}(u, v) = \frac{4\varepsilon_2}{N_{\text{outside}}}, \tag{5.6}$$

where the factor of 4 arises because each of the two markers has two ends, and N_{outside} is the number of pixels which may contain a marker end but which are not contained in an ROI. We define this as the number of pixels in the 256×256 region of the image described in Section 5.2 (representing the centre of the patient) which are not contained in an ROI.

5.5.4 All pixels in the projection

We shall now combine the expressions in Section 5.5.2 and 5.5.3 to derive the probability that any pixel in the projection image contains a marker end. With each marker appearing in an ROI with probability $(1 - \varepsilon_2)$, the probability that a particular ROI contains a marker is $2(1 - \varepsilon_2)/N_{\text{ROI}}$, where N_{ROI} is the number of ROIs in the image.

We assume that each ROI is equally likely to contain a marker. For each pixel within ROI $\mathcal{A}$, the probability that this pixel contains the end of marker is then

$$p^{\mathcal{A}}_{\text{inside}}(u,v) = 4(1-\varepsilon_2)\frac{1}{N_{\text{ROI}}}p^{\star}_{\mathcal{A}}(u,v), \tag{5.7}$$

which we can estimate by calculating

$$\hat{p}^{\mathcal{A}}_{\text{inside}}(u,v) = 4(1-\varepsilon_2)\frac{1}{N_{\text{ROI}}}\tilde{p}^{\star}_{\mathcal{A}}(u,v). \tag{5.8}$$

Finally, combining this with Equation 5.6, we have the estimated probability that pixel (u,v) contains a marker end:

$$\hat{\mathfrak{P}}((u,v)|y) = \begin{cases} p_{\text{outside}}(u,v)) & \text{if } (u,v) \text{ is not in any ROI} \\ \hat{p}^{\mathcal{A}}_{\text{inside}}(u,v) & \text{if } (u,v) \text{ is in ROI} \mathcal{A}. \end{cases} \tag{5.9}$$

Note that the sum of $\hat{p}^{\mathcal{A}}_{\text{inside}}$ over all pixels contained within any ROI is $4(1-\varepsilon_2)$, and that the sum of p_{outside} over all pixels not in an ROI is $4\varepsilon_2$. The sum over all pixels in the search space (whether inside an ROI or not) is therefore 4, the total number of marker ends.

At each pixel in the projection, we can now carry forward into our 3D analysis a probability. For pixels outside the 256×256 area where we expect to find markers, this is zero. For pixels where the ROI selection and sampler found a significantly positive probability, we take this almost at face value. For pixels where the ROI selection and sampler did not find a positive probability, we have allowed for error in those processes by assigning a low probability (based on ε_1 and ε_2).

The analysis in this section has been confined to one 2D projection. We write $\hat{\mathfrak{P}}_\theta((u,v)|y)$ for the value of $\hat{\mathfrak{P}}((u,v)|y)$ (Equation 5.9) calculated using the image data acquired at projection angle θ.

5.6 Combining the 2D results in 3D

Up until now, we have considered each projection in isolation, remarking that we can accept errors in the estimates in some images because the information from

each image will be combined in 3D space. In this section, we shall consider how to do this, using the estimates that each pixel in each 2D projection contains a marker end to produce a 3D probability map.

We have found the posterior distributions for the location of the marker ends in each of the 2D projection images, using a Bayesian model for the 2D projected markers. We seek to find the posterior probability that any point in 3D space is the end of a marker, given the data. Whilst our 2D model does not give us this directly, we can find an approximation. We shall now describe how we do this.

5.6.1 Combining posterior probabilities

Before we discuss the particular case of projection images of markers, we shall first consider a more general question: if we have several different sources of information, each of which gives us a posterior distribution, how can we combine the information from these different sources in a meaningful way?

Suppose X is a random variable which may take values in $A = \{a_1, \ldots, a_m\}$. Suppose we have data $Y = \{Y_1, \ldots, Y_n\}$, and we are given the posterior probabilities of X given each of the Y_i, $\pi(x|y_i)$.

Let us consider the product of these posterior probabilities,

$$f(x, y) = \prod_{i=1}^{n} \pi(x|y_i) \qquad \text{for } x \in A.$$

We now ask, what is the relationship between $f(x, y)$ and the posterior probability of X given all the data, $\pi(x|y)$?

If we knew the model (that is, the prior probability of X and the likelihood of $Y_i|X = a$), we could write $f(x, y)$ as

$$f(x, y) \propto [\pi(x)]^n \prod_{i=1}^{n} \pi(y_i|x) \qquad \text{for } x \in A. \tag{5.10}$$

We can now compare this to the posterior probability of X given all the data,

$$\pi(x|y) \propto \pi(x)\pi(y|x) \qquad \text{for } x \in A, \tag{5.11}$$

and ask under what circumstances is $f(x, y)$ (Equation 5.10) a good approximation to $\pi(x|y)$ (Equation 5.11)?

Suppose that the Y_i are conditionally independent given X. Then Equation 5.11 becomes

$$\pi(x|y) \propto \pi(x) \prod_{i=1}^{n} \pi(y_i|x) \qquad \text{for } x \in A. \tag{5.12}$$

Comparing this with Equation 5.10, we see that (up to proportionality), they are equal if

$$\pi(x) \propto [\pi(x)]^n \qquad \forall x \in A. \tag{5.13}$$

This is the case if $\pi(x)$ is constant for all x, i.e. if we have a flat prior.

The product of the $\pi(x|y_i)$ is therefore proportional to the posterior probability of X given all the data, $\pi(x|y)$, subject to two conditions:

- The Y_i must be conditionally independent given X.
- The prior $\pi(x)$ must be flat.

Application to marker data

In our case, we have the posterior distributions for the ends of the markers in the 2D projection images (Equation 5.9). If a marker end appears in a particular pixel (u, v) with probability η, we can say that the end of the 3D marker is in the patient on the line between the X-ray source and this pixel with probability η. We consider all points on this line to be equally probable locations for the marker end.

Our posterior distributions for the ends of the marker in 2D therefore give us posterior distributions for a new random variable: the location of the marker ends in 3D. These are posterior probabilities for the 3D location of the marker ends given the data in one projection image. We identify this 3D location with X in Section 5.6.1, with the data for one projection image being identified with Y_i; we have the posterior probabilities $\pi(x|y_i)$. We are interested in the posterior probability that a marker end is at a particular location in 3D space given the data in all the images, that is, $\pi(x|y)$.

Suppose we were to define a prior probability in 3D for the location of the marker ends. We note that in the selection of the ROIs and the subsequent analysis, we have not favoured any particular location in 3D, so we assume that our implied 3D prior is flat.

We now consider whether the projection data are independent. The 'noise' in the projection data comes from two sources: that due to the Poisson random processes (so-called 'quantum noise', as it is due to the interactions of individual photons), and so-called 'anatomical noise'—variations in the pixel value which are due to real structure in the patient, but which are not relevant to the particular question under consideration. The former of these may be expected to be independent between projections; the latter is not.

We note that we shall use only one in eight of the projection images; this means that the angles between projections is approximately 6°, and the structure of the patient may be expected to change markedly between projections. This suggests that the structural noise may be unimportant—if the quantum noise is the dominant noise source, the projections may be considered independent.

We shall therefore assume that the projections are independent. We shall consider the normalised product of the posterior probability given each projection to be an approximation to the posterior probability given the projection data in its entirety. In the next section, we shall describe the geometrical operations required to do this.

5.6.2 Determining the marker locations in 3D

Let $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) \in \mathbb{R}^3$ be a point within the patient. Let $\mathcal{P}_\theta$ be the projection operator, so that $(\mathfrak{u}, \mathfrak{v}) = \mathcal{P}_\theta(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ are the coordinates in projection θ to which the point $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ is projected. Let $p_{(u,v)} \subset \mathbb{R}^2$ be the area of the detector represented by pixel (u, v).

We have an estimate of the posterior probability that a pixel (u, v) in the projection at angle θ is the end of a marker, $\hat{\mathfrak{P}}_\theta((u, v)|y)$. For each $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ with $(\mathfrak{u}, \mathfrak{v}) = \mathcal{P}_\theta(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$, we shall use a multiple of $\hat{\mathfrak{P}}_\theta((u, v)|y)$ as an estimate of the probability that there is a marker at each of these $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$.

For each point $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$, we may calculate $\hat{\mathfrak{P}}_\theta((u, v)|y)$ for each projection angle θ. We consider the different values of θ to be analogous to the different values of i in the discussion in Section 5.6.1. We approximate the posterior probability that there is a marker end at 3D point $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ by the normalised product (over θ) of the posterior probabilities that there is a marker end in the pixel (u, v) such that $\mathcal{P}_\theta(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) \in p(u, v)$. That is, the probability density for a marker end being at point $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ is approximately

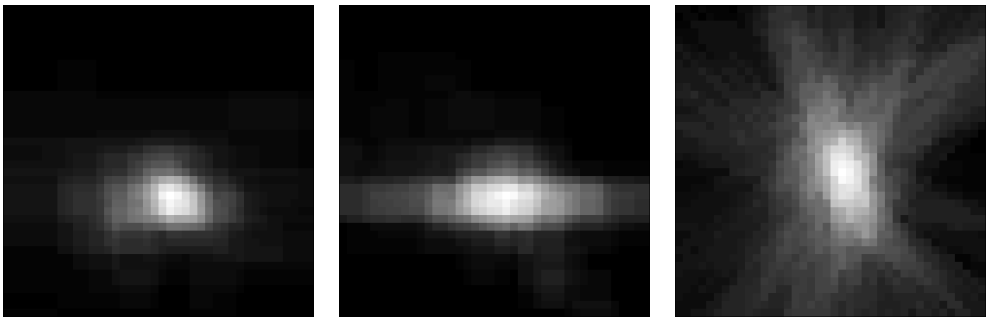
$$f(\text{marker end at } (\mathfrak{h}, \mathfrak{j}, \mathfrak{k})|y) \propto \prod_\theta \sum_{(u,v)} \hat{\mathfrak{P}}_\theta((u, v)|y)\mathbb{I}(\mathcal{P}_\theta(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) \in p_{(u,v)}). \quad (5.14)$$

We can calculate this probability density for points on a 3D lattice, and use these values to create credible volumes. In practice, we calculate the probabilities for each pixel in the projections on a log scale, which means we can conveniently use (unfiltered) back-projection to calculate, for each voxel, the sum of the log-probabilities of the pixels onto which the centre of that voxel is projected.

5.6.3 Experimental results

In this section, we shall show representative results for the 'bendy marker' data set. The results from the 'straight marker' data set were similar. For each experiment, we have used 77 projection images, which is one in eight of the total. We have used two different sets of 77: the first data set comprises images $1, 9, 17 \ldots$, while the second is images $4, 12, 20 \ldots$.

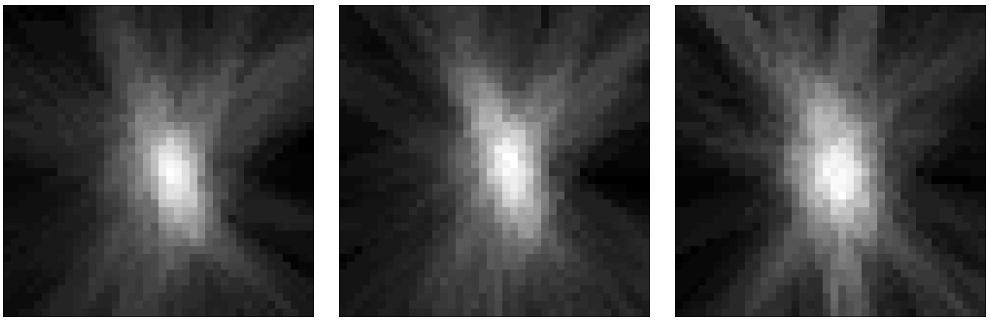
The morphological analysis and MCMC sampler were used, with markers proposed and visited during the 5000 iterations (after discarding the initial 2000), with the sampler run six times in each ROI. The probabilities for each pixel were assigned as described in Section 5.5, and combined in 3D as described in Section 5.6. The probabilities were calculated on a grid of 640^3 voxels of pitch 0.1mm. This was sufficient to cover the volume where the voxel probabilities were substantially different from the 'background' values (the sum of the log-probabilities assigned to pixels not in ROIs). The probabilities were renormalised such that their sum over the 640^3 volume was 4.



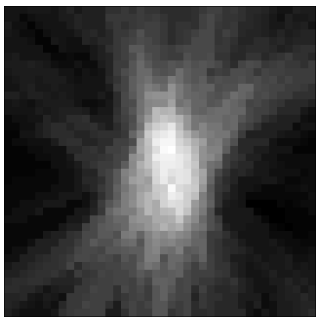
ɦ axis ɟ axis ɬ axis

Figure 5.35: Slices through a log-probability maximum.

The slices shown are perpendicular to the ɦ, ɟ and ɬ axes. The voxel pitch is 0.1mm.



Reconstructions from different runs.



Reconstruction from a different data set.

Figure 5.36: Reproducibility of the log-posterior probability.

The top row shows the log-probability calculated from three different experiments using the same set of data (data set 1). The bottom image shows the same slice from an experiment using data set 2.

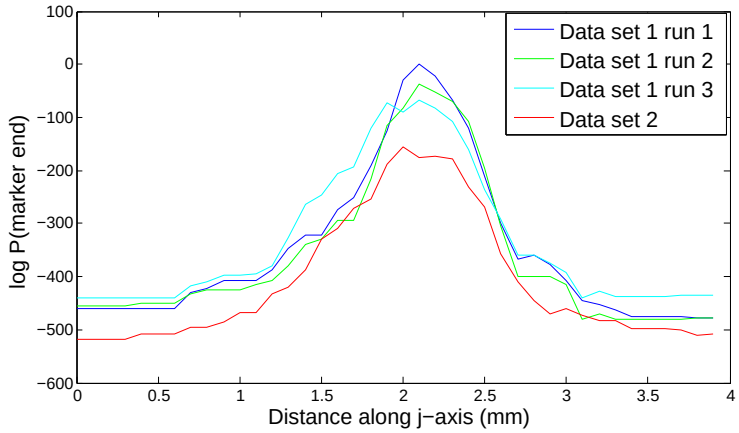


Figure 5.37: 1D profile of the log-probability after renormalising.

In each case, there were four clear maxima corresponding to the true marker end locations. Fig. 5.35 shows slices through one of these maxima, in planes perpendicular to the ɧ, j and ɬ directions. Fig. 5.36 shows the same slices, but with the results from different experiments and a different subset of data.

The voxel values (that is, the log-probability assigned to each point on the lattice) showed a large variation in magnitude, corresponding to a factor of more than 10^{200} difference in the probabilities of the most probable and least probable voxels. In fact, there was also a large variation in the probabilities of the four modes in each of the volumes, with the probability represented by the regions around the modes ranging from 10^{-40} to 4.00.

Fig. 5.37 shows the log-probability along the ɧ direction in the slices shown in Fig. 5.36, after renormalising such that the total probability in the region of this mode is 1.

We now attempt to create credible volumes for the marker ends in 3D; however, these are much smaller than expected. We shall illustrate this using the region around the mode shown in Figures 5.35 to 5.37. For the first experiment using data set 1, there is one voxel in the 3D neighbourhood with a (normalised) probability of $1 - 10^{-6}$. For sensible values of α, the credible volume for this marker end would therefore consist of this single voxel, with size $0.1\text{mm} \times 0.1\text{mm} \times 0.1\text{mm}$. For the second experiment using the same data, a 95% credible volume would include three voxels, with a 99% credible volume including five. There is no overlap between

the voxels included in the credible volumes calculated using the results of the two experiments—the single voxel from the first experiment is adjacent to the volume from the second experiment.

For the third experiment using the same data, we again have a single high-probability voxel (probability $1 - 10^{-5}$). In this case, the single-voxel credible volume coincides with the most probable voxel from the second experiment using data set 1. For the experiment using a different subset of the data (data set 2), the 99% credible volume consists of two voxels which do not appear in any of the other credible volumes, but which are adjacent to the credible volume from the second experiment using data set 1.

5.7 Removing the artefacts

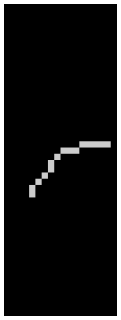
5.7.1 Removing the markers from the projections

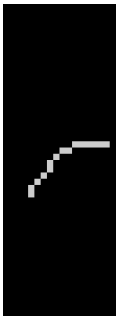
The primary purpose of the analysis presented in this chapter is to find the locations of the ends of the markers. While doing this, we can also produce a reconstruction which is free from the artefacts they cause. To do this, we determine which pixels in the projection images are likely to represent the marker, and replace them with a value representative of the 'background'.

Firstly, we need to determine which of our original ROIs we now believe to contain markers. We have determined the position of the marker ends in real 3D space. We can calculate the pixels in the projections into which they will be projected. We do this by using the projection operator, $\mathcal{P}_\theta$. For each 2D projection image, we calculate the four sets of indices (u, v) of the pixels into which the four end points are projected. We restrict further analysis to those ROIs which contain at least two projected marker ends.

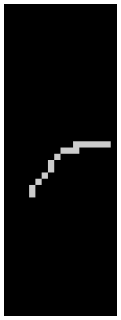
Recall that we used the MCMC sampler to create a list of possible locations for the pixelised markers in each ROI. We now truncate this list to a small number of the most probable. We have obtained good results using a simple method: retaining the two most probable. Each of the marker realisations in our truncated list has a set of marker pixels. We find the union of these sets, and create a binary image where the value of the marker pixels is one and the value of the other pixels

One of the ROIs.





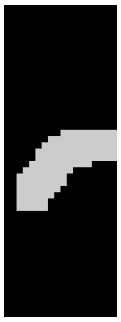
The two most probable markers in this ROI.



The union of the marker pixels for these two markers.

Figure 5.38: Selection of 'suspect' pixels.

5×5 pixel structuring element used to dilate the marker pixels.



The result of the dilation.

Figure 5.39: Dilation of the 'suspect' pixels.

is zero, as illustrated in Fig. 5.38. We perform the morphological dilation operation on this image, using a 5×5 pixel structuring element, illustrated in Fig. 5.39. We treat all the pixels with pixel value of one in this dilated image as 'suspect'—that is, these are the pixels we wish to replace with a different value.

The value we use is that obtained by calculating the morphological opening of the original projection image with the same 5×5 structuring element. Part of the projection image and the opening of the image are shown in Fig. 5.40.

Fig. 5.41 shows the image resulting from replacing the pixel values in the dilated marker with the values from the opened projection image. We can also produce images where the pixel value is a linear combination of that from the original image and from the image with the markers removed. Let $p_0(u, v)$ be the pixel

a: Part of the original projection image, showing the markers.

b: The opening of this image with the 5×5 structuring element shown in Fig. 5.39. The markers (and all other small bright features) have been removed.

Figure 5.40: Generation of 'background'.

value from the original image, and let $p_1(u, v)$ be that from the image with the markers removed. The pixel value in the new image is $\lambda p_0(u, v) + (1 - \lambda)p_1(u, v)$ for $0 \leq \lambda \leq 1$. Two such images are shown in Fig. 5.42. The markers are still visible, but less bright. The results of combining these images in an FDK reconstruction are shown in Section 5.7.2

5.7.2 Experimental results

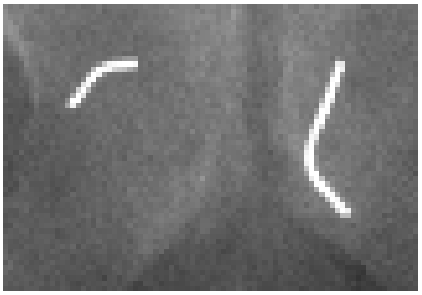
Figures 5.43 and 5.44 show axial slices from FDK reconstructions, implemented in C++. Each reconstruction uses 77 images (one in eight of the total data available). A 2D Hamming filter was applied to the image data before reconstruction. Results are shown using the original images ($\lambda = 1$), the images with markers removed ($\lambda = 0$), and linear combinations of these images with $\lambda = 0.1$ and $\lambda = 0.25$. The star artefacts visible in the reconstruction from the original images are much reduced or removed in the other images; the marker is visible (to a greater or lesser degree) in all the reconstructions except $\lambda = 0$.

Both markers are not always found in every image (either because the initial morphological analysis fails to identify an ROI containing a particular marker, or because the MCMC sampler fails to find it). This means that the artefact removal is not perfect. Figures 5.45 and 5.46 shows a different slice from the same reconstructions. Here, the artefacts have been reduced but not removed.

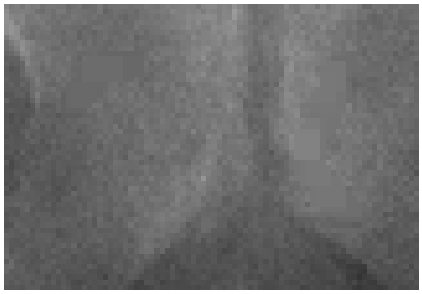
5.8 Discussion

The algorithm presented in this chapter reliably finds the ends of the markers in most cases (with the exception discussed below). When the voxel values representing the posterior probabilities in 3D are calculated, there are four clear modes, which, when plotted on the log scale (as in Fig 5.37) appears to have width of approximately 1mm. The location of the modes is consistent to within approximately 0.2mm between experiments conducted on the same subset of data, and when different subsets are used. The algorithm can find these locations reasonably quickly. We have also used these results to reduce (and in many cases remove) the streak and star artefacts.

a: The image with the markers removed.



b: A small region of the original image around the markers.



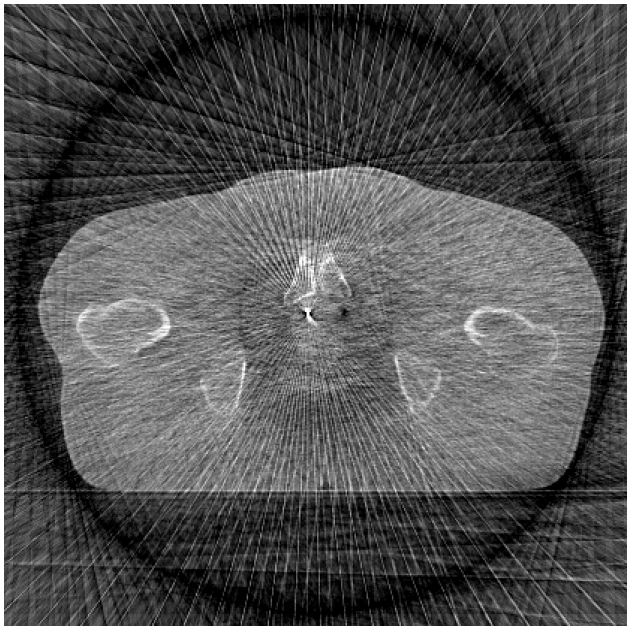
c: The same small region with markers removed.

Figure 5.41: The effect of removing the markers.

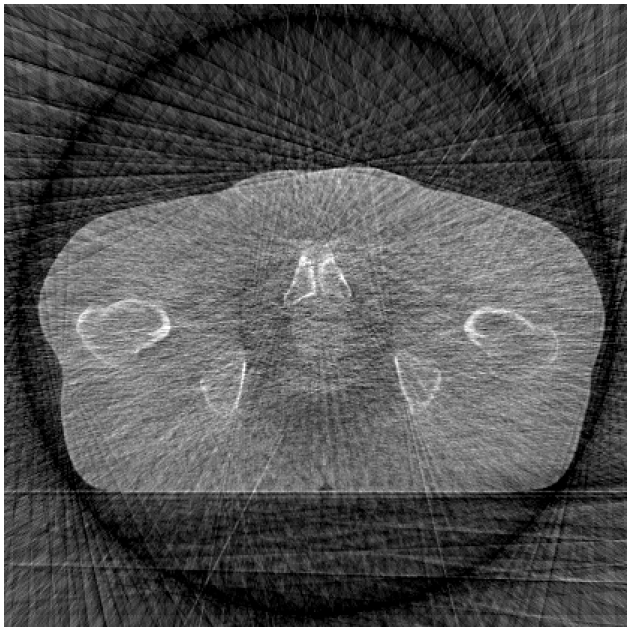
a: Image with $\lambda = 0.1$.

b: Image with $\lambda = 0.25$.

Figure 5.42: Linear combinations of the images.

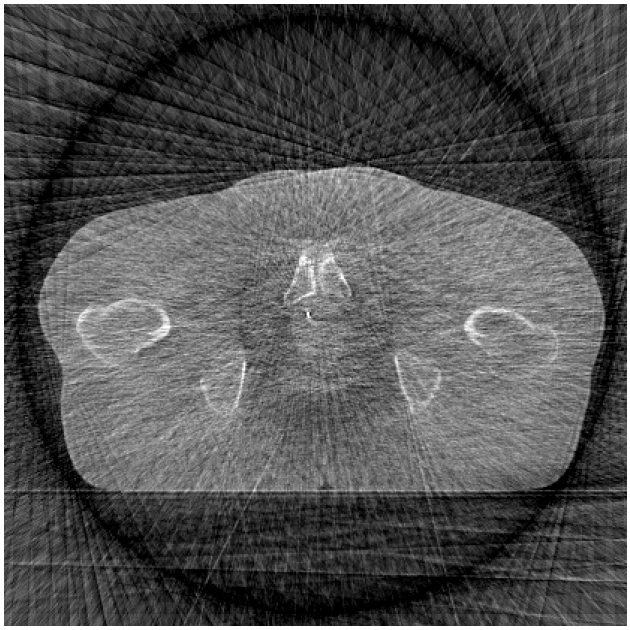


a: $\lambda = 1$ (original images)

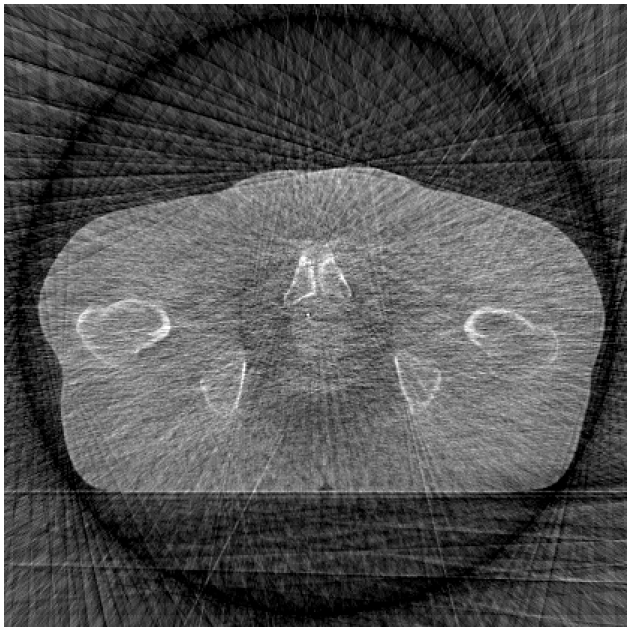


b: $\lambda = 0$

Figure 5.43: Reconstruction with artefacts removed.

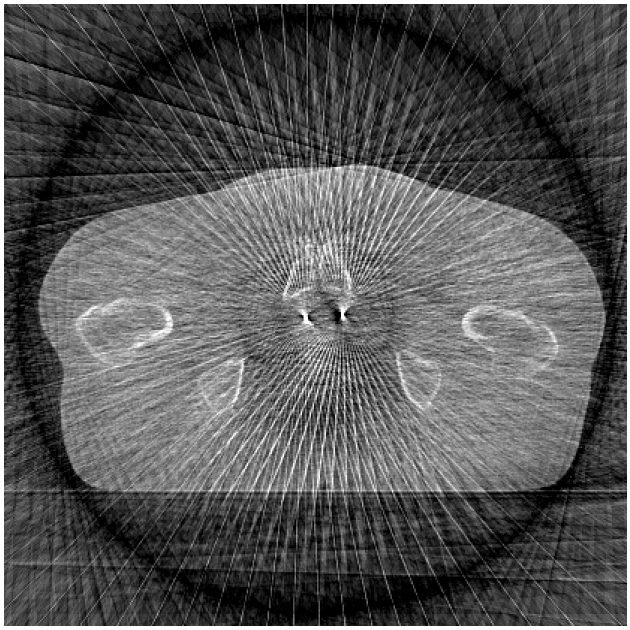


a: $\lambda = 0.25$

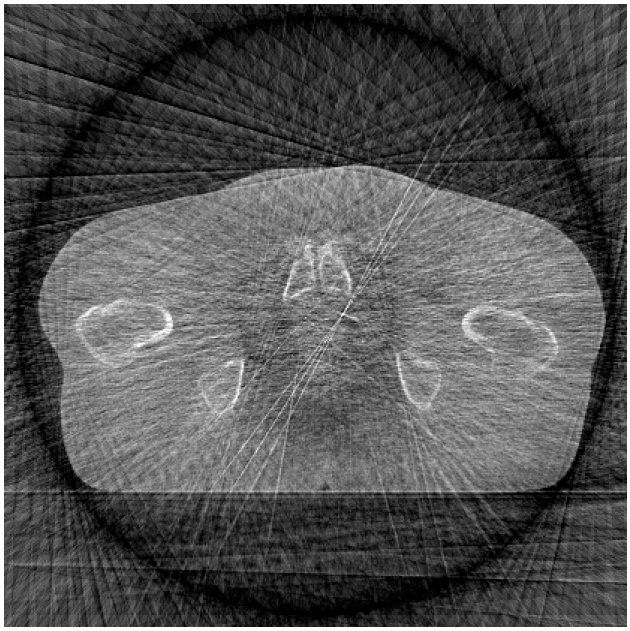


b: $\lambda = 0.1$

Figure 5.44: Reconstruction with artefacts removed, retaining the markers.

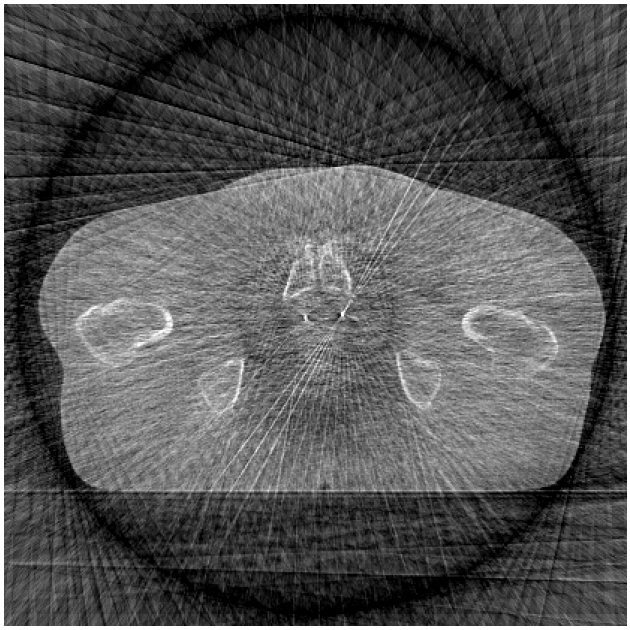


a: $\lambda = 1$ (original images)

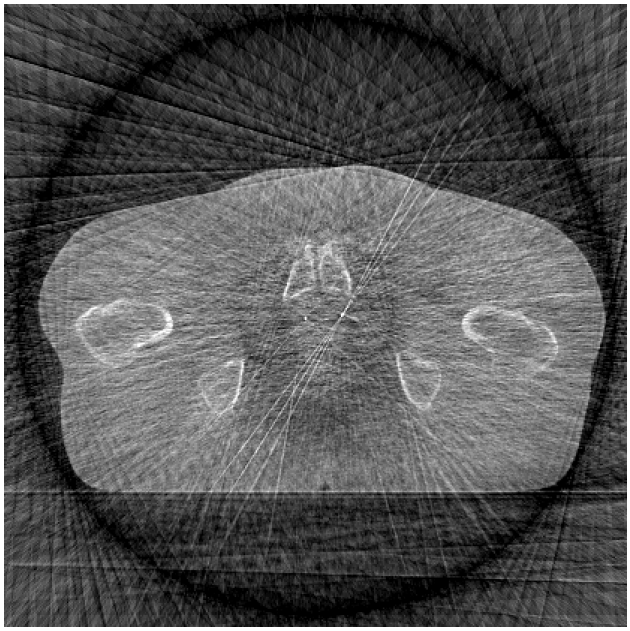


b: $\lambda = 0$

Figure 5.45: Reconstruction with artefacts reduced.



a: $\lambda = 0.25$



b: $\lambda = 0.1$

Figure 5.46: Reconstruction with artefacts reduced, retaining the markers.

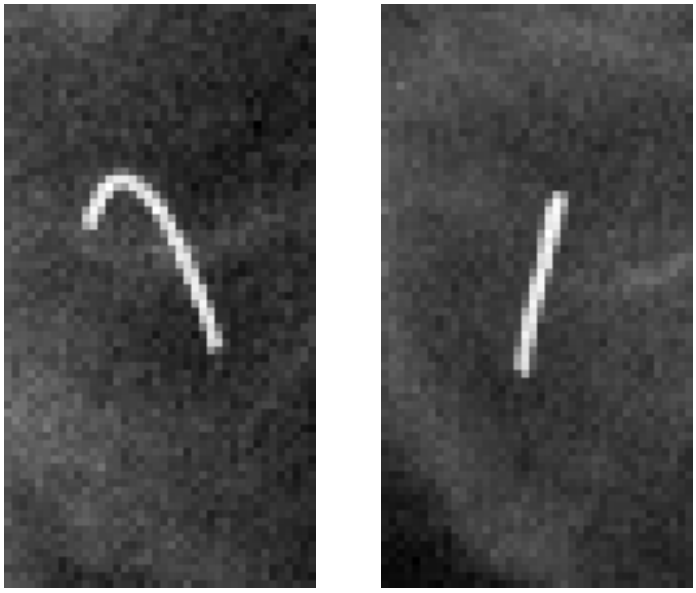


Figure 5.47: Two projections of the same 'hooked' marker.

However, we do not have a proper model for the location of the marker ends in 3D, given the data. We have demonstrated how we can approximate the posterior probability by multiplying the posterior probabilities from the individual projections. This approximation required two conditions: that the prior be flat and that the projections be independent. The second of these requirements is not met in our case—the noise in the images due to photon counting may be considered independent between projections, but differences in background pixel value due to the patient anatomy are not. In ignoring the dependencies between projections, we expect to see the resultant distribution too peaked, and, indeed, the credible volumes generated are very small—0.1 or 0.2mm in each direction.

In addition, the approach in this chapter cannot be used for 'hooked' markers, like the one illustrated in Fig 5.47. This figure shows two projections of the same marker, and in the right-hand image it is impossible to determine where one end of the marker is.

There are several ways the analysis demonstrated in this chapter could be developed:

- The model for the pixelised markers could be allowed to describe 'hooked' markers (with the hope that the end location would be correctly identified in sufficient projections that the ambiguity in others would not matter).
- Having found the modes representing the ends of the markers, new ROIs in the 2D images could be defined to include the points where these ends are projected. The sampler could be re-run (suitably modified to include two markers where one ROI includes two markers), to give a better estimate of the location of the ends. We would expect this to help particularly with the artefact removal, as more markers are identified correctly in projection images.

These improvements would not, however, address the fact that we do not have a proper model for the 3D marker, given the data. We have therefore not implemented them, choosing to focus instead on developing a proper model in 3D—the subject of the next chapter.

CHAPTER 6

MODELLING MARKERS IN 3D

6.1 Introduction

In the previous chapter, we demonstrated the utility of a model for the markers as they appear in 2D projection images. By sampling from the posterior distribution, we were able to describe the location of the markers in the projections. We then combined the information from many projections to describe the location of the markers in 3D space.

This is a computationally feasible approach, which is trivially parallelisable. However, it is not ideal: our model is of the pixelised 2D projected image of the marker, rather than of the 3D marker itself. This means that we do not have a proper model for the location of the markers in 3D space, given the data. Whilst we have had success in generating estimates of marker location and artefact-free reconstructions, we recognise that a series of 2D models for the projections of the markers will never provide as much information as a model for the markers in 3D. For example, with the 2D model there are problems with identifiability in projections where the two markers coincide, and when a marker bends back on itself in the $\mathbf{t}$-direction.

In this chapter, we shall examine a fully 3D model for the markers, and implement a scheme to sample from the posterior distribution which arises. The

model is more natural than that of the previous chapter, but the implementation is much more computationally demanding.

The attenuation properties of the patient's tissues and those of the implanted markers differ greatly. In this chapter, we shall consider the attenuation properties of the patient and the location of the markers as realisations of two separate random processes. The attenuation of the patient's tissues will be described on a voxel grid, for which we shall use a Markov random field model for the prior probability. We shall use a template model for the markers, for which we shall also define a prior probability distribution. We shall describe a forward model for the projection, which will be a probability distribution for the measured pixel values given the tissue attenuation values and marker locations.

We shall apply the method with a very simple template for a marker—a straight line in 3D space. This serves to illustrate how the model may be used, but is obviously not sufficient to model all situations. However, the model is conceived in such a way that more sophisticated templates would be easy to implement; this will be discussed later.

6.1.1 Notation

Let $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) \in \mathbb{R}^3$ be the Cartesian coordinates of real space. Without loss of generality, we shall assume that the voxel pitch is one (if not, the whole system can be rescaled). We consider the volume of real space represented by the coordinates $[h-\frac{1}{2}, h+\frac{1}{2}), [j-\frac{1}{2}, j+\frac{1}{2}), [k-\frac{1}{2}, k+\frac{1}{2})$ to be a voxel, labelled by the integer indices (h, j, k). Let $\mathfrak{N}_v = \{1, \ldots N_h\} \times \{1, \ldots N_j\} \times \{1, \ldots N_k\} \subset \mathbb{N}^3$ be the set of indices (h, j, k) which label the voxels. The total volume represented may be varied, but is typically $512 \times 512 \times 128$ voxels.

Similarly, let $(\mathfrak{u}, \mathfrak{v}) \in \mathbb{R}^2$ be the location of a point in the detector, in units such that incrementing $\mathfrak{u}$ or $\mathfrak{v}$ by one moves from a point in one pixel to the equivalent point in the next pixel. We consider the area of the detector represented by the coordinates $[u-\frac{1}{2}, u+\frac{1}{2}), [v-\frac{1}{2}, v+\frac{1}{2})$ to be a pixel, $p(u, v)$ for integer u and v. Each projection image is 512×512 pixels in size with pixel pitch 0.8mm (meaning that an increment of one in $\mathfrak{u}$ or $\mathfrak{v}$ represents an increment of 0.8mm.)

Let $\mathcal{P}_\theta : \mathbb{R}^3 \to \mathbb{R}^2$ be the projection operator, with $\mathcal{P}_\theta((\mathfrak{h}, \mathfrak{j}, \mathfrak{k})) = (\mathfrak{u}, \mathfrak{v})$ being the point in projection θ onto which real-space point $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ is projected.

6.2 Parameterisation of attenuation as two processes

In this section, we shall give a general description of the model we shall use. There are some subtleties in the details of what the random variables described here represent, which we shall discuss later.

We shall assume that the X-ray attenuation characteristics of the tissue within each voxel are constant, and we shall use a single (energy-independent) value to describe this. The 'attenuation' (in a loose sense which we shall discuss later) of the material in voxel (h, j, k) is denoted $X_{(h,j,k)}$. We write $X_{\text{tissue}} = \{X_{(h,j,k)}; (h, j, k) \in \mathfrak{N}_v\}$ for the array containing the value of $X_{(h,j,k)}$ for each (h, j, k).

Each marker is, in fact, a coil of very fine gold wire with the radius of the coil being 0.375mm. This is smaller than the voxel pitch (1mm), and too small for the hollow coiled structure to be resolved in the projection images. We therefore represent each marker as a cylinder in 3D space. The template for the markers consists of a curve representing the 'spine' of each marker. Let X_{marker} be the parameters of these two curves. For $i = 1, 2$, we define the function $M_i(t; x_{\text{marker}})$, with $M_i : [0, 1] \to \mathbb{R}^3$. The spine of the marker consists of the values of $M_i(t; x_{\text{marker}})$ for $t \in [0, 1]$.

For markers parameterised as two straight lines, we take X_{marker} to be the 3D coordinates of the end points. We shall write these as $X_{\text{marker}} = (S^{(1)}, S^{(2)}, S^{(3)}, S^{(4)})$. The individual scalar coordinates of $S^{(i)}$ will be denoted by $(S_1^{(i)}, S_2^{(i)}, S_3^{(i)})$. If we denote the reconstructed volume as $\mathcal{R} \subset \mathbb{R}^3$, the support of X_{marker} is $\mathcal{R}^4$. The paths of the central axes are given by

$$M_1(t) = S^{(1)} + t(S^{(2)} - S^{(1)}), \quad t \in [0, 1]$$

$$\text{and} \quad M_2(t) = S^{(3)} + t(S^{(4)} - S^{(3)}), \quad t \in [0, 1].$$

We shall denote the set of points on the central axis of marker i by M_i^c. The 3D cylinder representing each marker then consists of all points in the discs of radius

0.375mm, whose centre points are on the central axis of that marker and where the plane of the disc is perpendicular to that central axis. We shall denote the set of points in marker i by $M^{(i)}$.

We therefore have two sets of values, X_{tissue} and X_{marker}, about which we wish to infer. We shall write $X = (X_{\text{tissue}}, X_{\text{marker}})$ for the combined set of parameters.

6.3 Statistical model

We shall again employ a Bayesian approach. We shall consider the tissue attenuation and marker processes to be *a priori* independent and define a prior probability distribution for each. We shall propose a model for the measured projection data, Y, given $X = x$. This leads to a posterior distribution for X, given $Y = y$ which we shall use for inference on X.

6.3.1 Prior distribution of tissue attenuation values

We shall use a Markov random field to describe the prior probability of X_{tissue}. The value of voxel (h, j, k) is $X_{(h,j,k)}$. We shall write $(h_1, j_1, k_1) \sim (h_2, j_2, k_2)$ if (h_1, j_1, k_1) is a neighbour of (h_2, j_2, k_2), with $\mathfrak{C}$ being the set of cliques thus defined. We shall use a first order neighbourhood, with each voxel not at the edge of the reconstructed volume having six neighbours, each of which share a face with the central voxel. The neighbourhoods of voxels on the outside faces, edges and corners of the reconstructed volume contain five, four and three voxels respectively.

As in Chapter 4, we shall assign a higher probability to configurations where the difference in absolute value between neighbours is small (with our neighbourhoods now defined on a 3D voxel grid instead of the 2D pixel grid used in that chapter). We define the prior probability to be

$$\pi(x) \propto \exp\Big\{ -\sum_{c \in \mathfrak{C}} \Phi_c(x) \Big\}, \tag{6.1}$$

where

$$\Phi_{\mathfrak{c}}(x) = \beta \sum_{(h,j,k),(\tilde{h},\tilde{j},\tilde{k})\in\mathfrak{c}} \left| x_{(\tilde{h},\tilde{j},\tilde{k})} - x_{(h,j,k)} \right|,$$

with the parameter β controlling the correlation between neighbouring voxels. We again note, as in Chapter 4, that this is an improper prior (though the posterior will again be proper).

We now write $X_{\setminus(h,j,k)}$ to denote the set of all the voxel values except voxel (h,j,k). The conditional distribution of $X_{(h,j,k)}$ given $X_{\setminus(h,j,k)}$ is given by

$$\pi(x_{(h,j,k)}|x_{\setminus(h,j,k)}) \propto \exp\left\{ -\beta\left(\sum_{\substack{(\tilde{h},\tilde{j},\tilde{k}):\\ (\tilde{h},\tilde{j},\tilde{k})\sim(h,j,k)}} \left| x_{(\tilde{h},\tilde{j},\tilde{k})} - x_{(h,j,k)} \right| \right) \right\}.$$

6.3.2 Prior distribution of marker parameters

We shall now describe a joint probability distribution for the two markers. We know that one marker is approximately 10mm long and one approximately 20mm long. There is some uncertainty in the length due to the manufacturing process, the fact that the markers may compress and extend (recall that they are spring-like coils), and the fact that we are using a straight-line model to describe markers which are likely to be slightly bent. We use the length as the basis for our prior: a normal distribution on the length of the marker. We could include a parameter in our model to indicate which marker is the longer and which is the shorter, with the mean of the marginal distribution for each marker depending on whether it is the longer or shorter marker. This adds complexity to the sampling of the posterior distribution, so we instead use a distribution which is a mixture of normals, with means 10mm and 20mm and variance $\sigma^2 = 2^2$.

The distribution will be truncated, as the allowed lengths will be dependent on the size of the reconstructed volume and the locations of the end points within it. We shall also require that the markers are disjoint.

In Section 6.2, we gave the parameterisation of the two markers using their end points, $X_{\text{marker}} = (S^{(1)}, S^{(2)}, S^{(3)}, S^{(4)})$. The markers are implanted in such a way that they are close to vertical. In order to provide identifiability, we shall require that

the ɬ (cranio-caudal) coordinate of $S^{(1)}$ is greater than that of $S^{(2)}$ (i.e. $S_3^{(1)} > S_3^{(2)}$), and similarly we require $S_3^{(3)} > S_3^{(4)}$.

We can now write the prior probability of $X_{\text{marker}} = (S^{(1)}, S^{(2)}, S^{(3)}, S^{(4)})$ as

$$\pi(x_{\text{marker}}) = \begin{cases} C\,\mathbb{I}_{\text{disjoint}}(x_{\text{marker}})\mathbb{I}_{\text{oriented}}(x_{\text{marker}})f_1(x_{\text{marker}})f_2(x_{\text{marker}}) \\ \qquad\qquad \text{for } x_{\text{marker}} = (s^{(1)}, s^{(2)}, s^{(3)}, s^{(4)}), s^{(i)} \in \mathcal{R} \\ 0 \qquad\qquad \text{otherwise} \end{cases} \tag{6.2}$$

where

$$\mathbb{I}_{\text{disjoint}}(x_{\text{marker}}) = \begin{cases} 1 & \text{for } m^{(1)}(x_{\text{marker}}) \cap m^{(2)}(x_{\text{marker}}) = \emptyset \\ 0 & \text{otherwise,} \end{cases}$$

$$\mathbb{I}_{\text{oriented}}(x_{\text{marker}}) = \begin{cases} 1 & \text{for } s_3^{(1)} > s_3^{(2)} \text{ and } s_3^{(3)} > s_3^{(4)}, \\ & \qquad \text{where } x_{\text{marker}} = (s^{(1)}, s^{(2)}, s^{(3)}, s^{(4)}) \\ 0 & \text{otherwise,} \end{cases}$$

$$\begin{aligned} f_1(x_{\text{marker}}) =& \frac{-1}{2\sqrt{2\pi\sigma^2}} \exp\left\{\frac{-1}{2\sigma^2}(|s^{(2)} - s^{(1)}| - 10)^2\right\} \\ &+ \frac{1}{2\sqrt{2\pi\sigma^2}} \exp\left\{\frac{1}{2\sigma^2}(|s^{(2)} - s^{(1)}| - 20)^2\right\}, \end{aligned}$$

$$\begin{aligned} f_2(x_{\text{marker}}) =& \frac{-1}{2\sqrt{2\pi\sigma^2}} \exp\left\{\frac{-1}{2\sigma^2}(|s^{(4)} - s^{(3)}| - 10)^2\right\} \\ &+ \frac{1}{2\sqrt{2\pi\sigma^2}} \exp\left\{\frac{1}{2\sigma^2}(|s^{(4)} - s^{(3)}| - 20)^2\right\}, \end{aligned}$$

and C is a normalising constant.

6.3.3 Likelihood of data

The data Y are the 600 or so projection images (indexed by θ), each one of which contains 512×512 pixels (indexed by (u, v)). We shall assume that the projected pixel values $Y_{(u,v,\theta)}$ are conditionally independent given X.

As elsewhere in this thesis, we shall use a normal likelihood: $Y_{(u,v,\theta)}|x \sim \mathcal{N}(g(u, v, \theta; x), \sigma^2)$. We shall describe in this section how we define the function $g(u, v, \theta; x)$, which defines the mean of this distribution. We shall also define exactly what we mean by X_{tissue} representing 'attenuation', as in Section 6.2.

Let $I_0(\mathcal{E}, (\mathfrak{u}, \mathfrak{v}))$ be the raw signal that would be measured in the absence of any attenuation, due to photons at energy $\mathcal{E}$ incident at point $(\mathfrak{u}, \mathfrak{v})$ in the detector. Ignoring scattered radiation, the mean raw signal measured in pixel (u, v) in the absence of any attenuation is

$$I_0(u, v) = \int_{u-\frac{1}{2}}^{u+\frac{1}{2}} \int_{v-\frac{1}{2}}^{v+\frac{1}{2}} \int_{\mathcal{E}} I_0(\mathcal{E}, (\mathfrak{u}, \mathfrak{v})) \, \mathrm{d}\mathcal{E} \, \mathrm{d}\mathfrak{v} \, \mathrm{d}\mathfrak{u}.$$

The effect of the attenuating materials is energy-dependent. Let $z^{(\mathcal{E})}(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ be the linear attenuation at point $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ at energy $\mathcal{E}$. The raw signal $I(u, v)$ measured in the presence of attenuation is then given by

$$I(u, v) = \int_{u-\frac{1}{2}}^{u+\frac{1}{2}} \int_{v-\frac{1}{2}}^{v+\frac{1}{2}} \int_{\mathcal{E}} I_0(\mathcal{E}, (\mathfrak{u}, \mathfrak{v})) \exp\left\{ - \int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z^{(\mathcal{E})}(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) \mathrm{d}l \right\} \mathrm{d}\mathcal{E} \, \mathrm{d}\mathfrak{v} \, \mathrm{d}\mathfrak{u}, \quad (6.3)$$

where $\mathcal{L}_\theta(\mathfrak{u}, \mathfrak{v})$ is the path from the X-ray source to point $(\mathfrak{u}, \mathfrak{v})$ in the detector in the projection at angle θ.

We shall represent the total attenuation $z^{(\mathcal{E})}(\mathfrak{h}, \mathfrak{j}, \mathfrak{k})$ by two components: one due to the tissues and one due to the markers. We write

$$z^{(\mathcal{E})}(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) = z^{(\mathcal{E})}_{\text{tissue}}(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) + z^{(\mathcal{E})}_{\text{marker}}(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}).$$

We assume that the attenuation of the tissues is constant within each voxel. We can therefore write

$$\int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z^{(\mathcal{E})}_{\text{tissue}}(\mathfrak{h},\mathfrak{j},\mathfrak{k})\mathrm{d}l \approx \sum_{(h,j,k)} \delta_{((\mathfrak{u},\mathfrak{v},\theta),(h,j,k))} z^{(\mathcal{E})}_{\text{tissue}}(h,j,k),$$

where $\delta_{((\mathfrak{u},\mathfrak{v},\theta),(h,j,k))}$ is the length of the intersection of the path $\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})$ with voxel (h,j,k).

We now write Equation 6.3 as

$$I(u,v) \approx \int_{u-\frac{1}{2}}^{u+\frac{1}{2}} \int_{v-\frac{1}{2}}^{v+\frac{1}{2}} \int_{\mathcal{E}} I_0(\mathcal{E},(\mathfrak{u},\mathfrak{v})) \exp\left\{ -\left(\int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z^{(\mathcal{E})}_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k})\mathrm{d}l \right) - \sum_{(h,j,k)} \delta_{((\mathfrak{u},\mathfrak{v},\theta),(h,j,k))} z^{(\mathcal{E})}_{\text{tissue}}(h,j,k) \right\} \mathrm{d}\mathcal{E}\, \mathrm{d}\mathfrak{v}\, \mathrm{d}\mathfrak{u}. \tag{6.4}$$

We now make another approximation: that the attenuation of the marker $z^{(\mathcal{E})}_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k})$ is constant with respect to $\mathcal{E}$, and write its value as $z_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k})$. We make the same approximation for $z^{(\mathcal{E})}_{\text{tissue}}(h,j,k)$, writing its value as $z_{\text{tissue}}(h,j,k)$.

Equation 6.4 now becomes

$$I(u,v) \approx \int_{u-\frac{1}{2}}^{u+\frac{1}{2}} \int_{v-\frac{1}{2}}^{v+\frac{1}{2}} I_0(\mathfrak{u},\mathfrak{v}) \exp\left\{ -\left(\int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k})\mathrm{d}l \right) - \sum_{(h,j,k)} \delta_{((\mathfrak{u},\mathfrak{v},\theta),(h,j,k))} z_{\text{tissue}}(h,j,k) \right\} \mathrm{d}\mathfrak{v}\, \mathrm{d}\mathfrak{u}. \tag{6.5}$$

Our next task in calculating an approximation to this quantity is to avoid the need to integrate over the area of the pixel (u,v). We achieve this in different ways for the two components of the model.

Tissue attenuation

The values of the tissue attenuation do not change markedly from one voxel to another. It is therefore a reasonable approximation to write

$$\sum_{(h,j,k)} \delta_{((\mathfrak{u},\mathfrak{v},\theta),(h,j,k))} z_{\text{tissue}}(h,j,k)$$

$$\approx \sum_{(h,j,k)} \delta_{((u,v,\theta),(h,j,k))} z_{\text{tissue}}(h,j,k) \qquad \text{for } (\mathfrak{u},\mathfrak{v}) \text{ in pixel } (u,v).$$

Similarly, we assume that $I_0(\mathfrak{u},\mathfrak{v})$ is constant within pixel (u,v), and take its value as $I_0(u,v)$.

We now substitute this into Equation 6.5, obtaining

$$I(u,v) \approx I_0(u,v) \int_{u-\frac{1}{2}}^{u+\frac{1}{2}} \int_{v-\frac{1}{2}}^{v+\frac{1}{2}} \exp\left\{ - \int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k}) \mathrm{d}l \right\} \mathrm{d}\mathfrak{v}\, \mathrm{d}\mathfrak{u}$$

$$\times \exp\left\{ - \sum_{(h,j,k)} \delta_{((u,v,\theta),(h,j,k))} z_{\text{tissue}}(h,j,k) \right\} \tag{6.6}$$

(noting that, by the definition of the units of $\mathfrak{u}$ and $\mathfrak{v}$, the area of a pixel is one square unit).

We define

$$g'_{\text{tissue}}(z_{\text{tissue}}; u, v, \theta) = \sum_{(h,j,k)} \delta_{((u,v,\theta),(h,j,k))} z_{\text{tissue}}(h,j,k), \tag{6.7}$$

so that Equation 6.6 becomes

$$I(u,v) \approx I_0(u,v) \int_{u-\frac{1}{2}}^{u+\frac{1}{2}} \int_{v-\frac{1}{2}}^{v+\frac{1}{2}} \exp\left\{ - \int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k}) \mathrm{d}l \right\} \mathrm{d}\mathfrak{v}\, \mathrm{d}\mathfrak{u}$$

$$\times \exp\left\{ - g'_{\text{tissue}}(z_{\text{tissue}}; u, v, \theta) \right\}. \tag{6.8}$$

To turn the raw signal into a projection value, the signal is scaled and log transformed. We obtain

$$\begin{aligned} p_\theta(u,v) &= -\log\left(\frac{I(u,v)}{I_0(u,v)}\right) \\ &\approx g'_{\text{tissue}}(z;u,v,\theta) \\ &\quad + \log\left[\int_{u-\frac{1}{2}}^{u+\frac{1}{2}}\int_{v-\frac{1}{2}}^{v+\frac{1}{2}} \exp\left\{-\int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k})\mathrm{d}l\right\}\mathrm{d}\mathfrak{v}\,\mathrm{d}\mathfrak{u}\right]. \end{aligned} \tag{6.9}$$

Marker attenuation

The signal in pixel (u,v) is affected by any material present in the pyramid which has the pixel as its base and the X-ray source as its apex. We have simplified the calculation of the effect of the tissue attenuation (the attenuation of the patient without the markers), so that we only consider the attenuation along the central axis of this pyramid.

We now turn our attention to the attenuation due to the marker. We wish to find an approximation to

$$Z(u,v) = \log\left[\int_{u-\frac{1}{2}}^{u+\frac{1}{2}}\int_{v-\frac{1}{2}}^{v+\frac{1}{2}} \exp\left\{-\int_{\mathcal{L}_\theta(\mathfrak{u},\mathfrak{v})} z_{\text{marker}}(\mathfrak{h},\mathfrak{j},\mathfrak{k})\mathrm{d}l\right\}\mathrm{d}\mathfrak{v}\,\mathrm{d}\mathfrak{u}\right]. \tag{6.10}$$

We can think of the markers as casting a 'shadow' on the pixels. A pixel might be entirely covered by a marker, or the marker might only cast a shadow in a corner. If we only considered the central axis of the pyramid, as we are doing for the tissue attenuation, we would not include any contribution to the pixel value from a marker casting its shadow in a corner of the pixel only. While the central axis approximation is reasonable for the rest of the patient (where the attenuation properties do not generally change greatly from voxel to voxel), it does not suffice for the markers (where there are very rapid changes, so that the attenuation along the path to the centre of a pixel may be very different from that along the path to a point elsewhere in the pixel).

In this thesis, we are not attempting to model the hollow coiled structure of the markers—we model them as simple cylinders. Rather than using the true linear attenuation coefficient for gold, we consider

$$z_{\text{marker}}(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) = \mathbb{I}(\text{marker at } (\mathfrak{h}, \mathfrak{j}, \mathfrak{k})) \times w,$$

where w is the marker 'weight', which we shall choose empirically.

In order to calculate the value of $Z(u, v)$ we need to know the path length through the marker to each point in pixel (u, v). We can find an approximation to this by simulating points in our modelled marker, which is a cylinder of radius $r = 0.375$mm in 3D space.

We shall demonstrate the approach using a cylinder of length $l = 10.0$mm, with the vector representing the central axis of the cylinder being $(-1.4, 2.8, 9.5)$. The cylinder is aligned closest to the $\mathfrak{k}$ axis, but not exactly parallel to any axis.

$N = 10^8$ 3D points $(\mathfrak{h}_i, \mathfrak{j}_i, \mathfrak{k}_i)$ were simulated within this cylinder. For this, we used a stratified sampling approach. We sample a point in each of the N discs whose central points are evenly spaced along the central axis of the cylinder. Within each disc, the point is sampled uniformly by area. The geometry from one of the real projection images was used to define a projection operator $\mathcal{P}_\theta$. The projected 2D coordinates $(\mathfrak{u}, \mathfrak{v})$ of each of the N 3D points were calculated, with $(\mathfrak{u}_i, \mathfrak{v}_i) = \mathcal{P}_\theta(\mathfrak{h}_i, \mathfrak{j}_i, \mathfrak{k}_i)$.

We shall now use very small pixels to find an approximation to the path lengths through the marker. Suppose the pixel $p(n_u, n_v)$ represents the area of the detector $[u_{n_u} - \Delta/2, u_{n_u} + \Delta/2) \times [v_{n_v} - \Delta/2, v_{n_v} + \Delta/2)$, where Δ is the pixel pitch. Let $\ell_{(n_u, n_v)}$ be the length of the path through the marker to the centre of pixel (n_u, n_v). We approximate $\ell_{(n_u, n_v)}$ by the estimated volume of the part of the marker within the pyramid defined by the pixel, divided by the area of the pixel. The volume is estimated by counting the number of points $(\mathfrak{h}_i, \mathfrak{j}_i, \mathfrak{k}_i)$ with $\mathcal{P}_\theta(\mathfrak{h}_i, \mathfrak{j}_i, \mathfrak{k}_i) = (\mathfrak{u}_i, \mathfrak{v}_i) \in p(n_u, n_v)$. That is,

$$\hat{\ell}_{(n_u, n_v)} \approx C \sum_i \mathbb{I}[(\mathfrak{u}_i, \mathfrak{v}_i) \in p(n_u, n_v)],$$

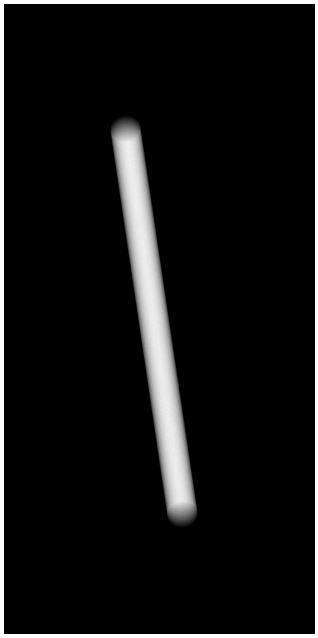


Figure 6.1: Simulated projected marker.

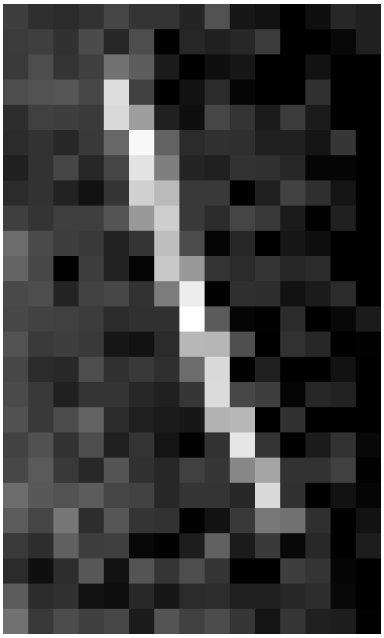
where

$$C = \frac{\pi r^2 l}{N\Delta^2}.$$

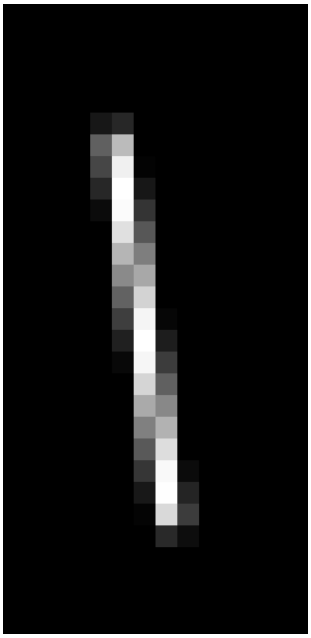
We consider the case where there is a uniform background, which we shall assume to be zero. The measured signal in the small pixel (n_u, n_v) is then approximately proportional to $\exp\{-w\ell_{(n_u,n_v)}\}$, where w is the marker weight. The smaller the pixel pitch Δ the more accurate the approximation, but with a larger number of simulated points N required. Fig. 6.1 shows a simulated projection of the marker, with $\Delta = 0.016$mm.

Having approximated the path-lengths using these small pixels, we can calculate an approximation to the pixel values we would obtain in the real projection geometry, where $\Delta = 0.8$mm. We call this the fine-mesh approximation. The pixel value in pixel (u, v) is given by

$$\mu_{(u,v)} = \log \sum_{p(n_u,n_v)\subset p(u,v)} \exp\{-w\ell_{(n_u,n_v)}\}.$$



Close-up of a real projection image showing the projection of a marker.



The mean pixel values generated using the fine-mesh approximation to the projection process.

Figure 6.2: Illustration of fine-mesh approximation to the markers.
Note that the markers are not intended to be in the same spatial location.

We are not trying to model the true linear attenuation of the markers, so we can choose the value of w which gives us a projected marker which matches the real data in terms of the pixel value. Fig. 6.2 shows part of real projection image, and a simulated projection image, with w chosen so that the pixel values in the images match.

We have calculated this approximation by calculating many path-lengths per pixel, at a resolution of 0.016mm. We could use an even finer mesh (in which case the approximation would be more accurate, as long as N was big enough), or a coarser mesh (in which case the approximation will be less accurate). The limiting case is a resolution of 0.8mm (the true pixel pitch), so that we are using only one estimate of the path length per pixel.

It turns out that using only one estimate per pixel provides a good approximation. This is shown in Fig. 6.3. In effect, we are using the sum to approximate the logarithm of the sum of exponentials.

To be more explicit, the approximation we use in our model is a coarse-mesh approximation. We calculate the contribution to the pixel value due to the marker in the same way that we calculated the values on the fine mesh. That is, we simulate some number N of 3D points in the marker $(\mathfrak{h}_i, \mathfrak{j}_i\mathfrak{k}_i)$ and then calculate the contribution from the marker as

$$Z \approx g'_{\text{marker}}(z; u, v, \theta) = w \sum_i \mathbb{I}[\mathcal{P}_\theta(\mathfrak{h}_i, \mathfrak{j}_i\mathfrak{k}_i) \in p(u, v)]. \tag{6.11}$$

The value of w is chosen so that the simulated pixel values match those in the real data. Fig. 6.4 shows the experiment repeated with a different value w. It is clear that the linear approximation would not be appropriate in this case.

We now write Equation 6.9 as

$$p_\theta(u, v) \approx g'_{\text{tissue}}(z; u, v, \theta) + g'_{\text{marker}}(z; u, v, \theta). \tag{6.12}$$

Relationship between data and attenuation values

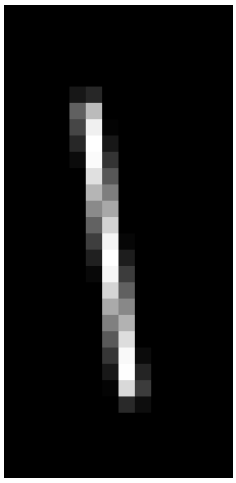
We have previously alluded to the fact that the values we denote X are not the true attenuation values. In this section, we shall describe the relationship between the linear attenuation coefficients $z_{\text{tissue}}(h, j, k)$ and the voxel values $x_{\text{tissue}}(h, j, k)$.

We have given expressions for the projection values,

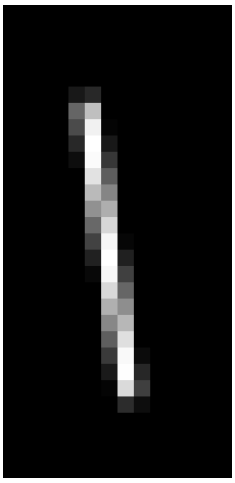
$$\begin{aligned} p_\theta(u, v) &= -\log\left(\frac{I(u, v)}{I_0(u, v)}\right) \\ &\approx g'_{\text{tissue}}(z; u, v, \theta) + g'_{\text{marker}}(z; u, v, \theta). \end{aligned}$$

We consider the data to be realisations of a random variable Y, whose expectation is related to these projection values. As described in Chapter 2, the pixel values are a scaled and centred version of these projection values. We therefore write

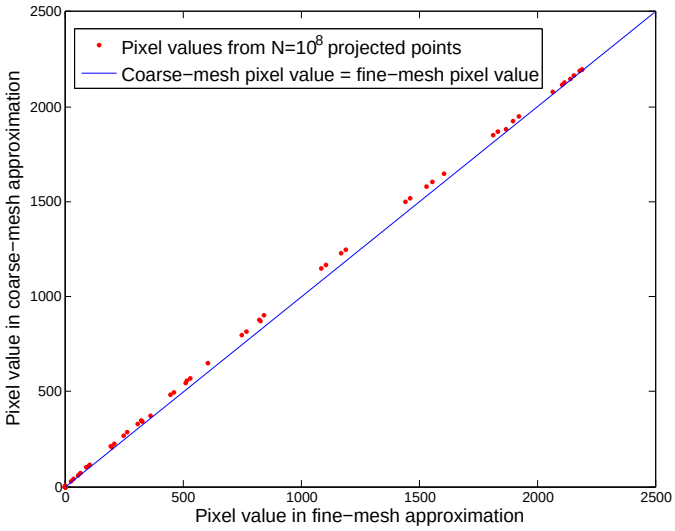
$$\mathbb{E}(y_{(u,v)}) = a p_\theta(u, v) + b,$$



The mean pixel values generated using the fine-mesh approximation to the projection process.



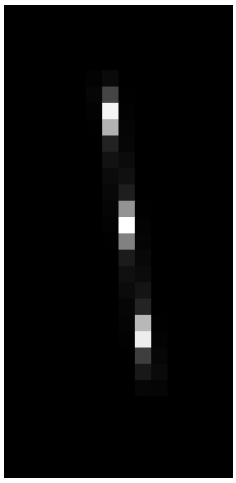
The mean pixel values generated using the coarse-mesh approximation to the projection process.



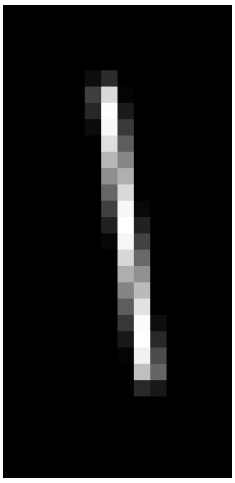
Scatter plot showing the relationship between the pixel values.

Figure 6.3: The coarse-mesh approximation to the markers.

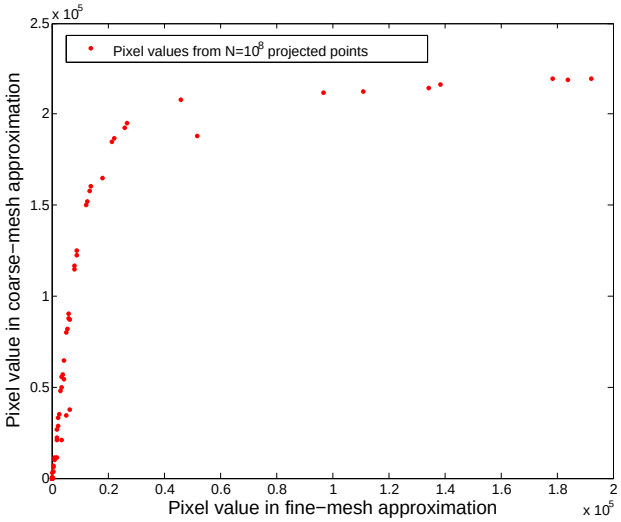
The figure compares the fine-mesh and coarse-mesh approximations to the markers.



The mean pixel values generated using the fine-mesh approximation to the projection process.



The mean pixel values generated using the coarse-mesh approximation to the projection process.



Scatter plot showing the relationship between the pixel values.

Figure 6.4: Coarse-mesh approximation with a different weighting.

The figure compares the fine-mesh and coarse-mesh approximations to the markers, with a different weighting to that used in Fig. 6.3

where a and b are arbitrary and unknown constants.

We define the voxel values X such that

$$\mathbb{E}(y_{(u,v)}) = g(x; u, v, \theta) = g_{\text{tissue}}(x; u, v, \theta) + g_{\text{marker}}(x; u, v, \theta),$$

where

$$g_{\text{tissue}}(x; u, v, \theta) = \sum_{(h,j,k)} \delta_{((u,v,\theta),(h,j,k))} x_{\text{tissue}}(h, j, k)$$

and

$$g_{\text{marker}}(x; u, v, \theta) = w \sum_i \mathbb{I}[\mathcal{P}_\theta(\mathfrak{h}_i, \mathfrak{j}_i \mathfrak{k}_i) \in p(u, v)].$$

Our definition of $g_{\text{marker}}(x; u, v, \theta)$ thus includes empirical scaling parameters such that the pixel values predicted by our model match those seen in the data. The tissue attenuation values X_{tissue} are not absolute values—to obtain values in particular units (e.g. mm^{-1}), materials with known linear attenuation coefficient (such as Perspex and PTFE) would need to be included in the scan volume.

We use a normal likelihood:

$$y_i | x \sim \mathcal{N}(g(u, v, \theta; x), \sigma^2). \tag{6.13}$$

We note that allowing the definition of the tissue attenuation values to depend on the data is a rather empirical approach, and in particular that defining a prior based on the absolute differences between these values is problematic. However, we also note that we are determining many of the parameters in this model empirically (for example, the parameter β in the prior for the tissue attenuation values). We find, in practice, that the range of a and b values encountered is low compared to the changes in the empirical parameters required to change the behaviour of the model.

6.4 Analysing the posterior distribution of X given Y

In Equations 6.1 and 6.2, we have defined prior distributions for X_{tissue} and X_{marker}. We assume that these are independent random variables, and so the prior distribution for $X = (X_{\text{tissue}}, X_{\text{marker}})$ is the product of the two. Combined with the likelihood for the data Y given X (Equation 6.13), this defines the posterior distribution of X given Y.

We may instead choose not to model X_{marker}, including only X_{tissue} in the model. We then consider the posterior distribution $\pi(X_{\text{tissue}}|y)$. We do this by setting $g_{\text{marker}}(X; u, v, \theta) = 0$ (so the only attenuation is due to tissues), and by setting the prior probability of the (non-existent) markers to a constant.

On the other hand, we may include only X_{marker} as a random part of the model, and consider the posterior distribution $\pi(X_{\text{marker}}|y)$. In this case, we cannot set $g_{\text{tissue}}(X; u, v, \theta) = 0$, as we need to know the 'background' pixel value to which we add the contribution from the marker. Instead, we set $g_{\text{tissue}}(X; u, v, \theta)$ to some fixed value. In the case of the phantom, this could be the truth or a reconstruction of the same phantom without markers; with real patient data we shall find other approaches to this problem.

As before, we shall use a Metropolis-Hastings sampler to analyse the posterior distribution. If we are modelling the tissue attenuation, we update the values in X_{tissue} in a raster fashion. If we are modelling the markers, we choose one of the twelve scalar parameters to update. If we are modelling both, the sampler alternates between doing one sweep of the tissue attenuation values and updating one parameter of the markers.

The remainder of this chapter will describe how this sampler has been implemented, and demonstrate some results from running the sampler.

6.4.1 Implementation

The sampler and other associated routines are more computationally intensive than that described in previous chapters. For this reason, the software used in this chapter was implemented in C++.

For each iteration of the Metropolis-Hastings sampler, we shall update the tissue attenuation at each voxel, $x_{\text{tissue}}(h,j,k)$ in turn. We shall then update the position of the marker ends, x_{marker}.

Updating the tissue attenuation values

To update the tissue attenuation value for a voxel, we calculate the likelihood ratio. For this we need to know the measured pixel value $y_{(u,v,\theta)}$ for each pixel where $\delta_{((u,v,\theta),(h,j,k))} \neq 0$ (note that this is at least one pixel per projection). We also need to know the path lengths and voxel values for all other voxels on this ray in order to calculate $g_{\text{tissue}}(x;u,v,\theta)$.

We pre-calculate the $\delta_{(\cdot)}$ values. We start by considering each projection in turn. For each pixel (u,v,θ), we calculate the positions in real space of the X-ray source and the centre of the pixel, and ray-trace between these points using Han *et al*'s variation of Siddon's algorithm [18]. We create a data structure which we can access using the voxel indices (h,j,k). For each (h,j,k), we store a list of the pixel indices (u,v,r_θ) and the path lengths $\delta_{(\cdot)}$ for each pixel where $\delta_{((u,v,\theta),(h,j,k))} \neq 0$.

In order to calculate the likelihood ratio for the proposed and current values of voxel (h,j,k), we look up this voxel in our data structure. This gives us a list of pixels $\{(u,v,r_\theta)\}$ and a list of path lengths $\{\delta_{(\cdot)}\}$ to those pixels. The exponent in the normal likelihood is

$$\frac{1}{2\sigma^2}\sum_{(u,v,r_\theta)}\left[g_{\text{tissue}}(x_{\text{tissue}};u,v,\theta)-y_{(u,v,\theta)}\right]^2.$$

We update one element of x_{tissue} at a time. Let the proposed value be $x^{(\star)}_{(h,j,k)}$, and the current value (at iteration r) be $x^{(r)}_{(h,j,k)}$. The values of $g_{\text{tissue}}(x_{\text{tissue}};u,v,\theta)$ for the proposed and current values are related by

$$\begin{aligned} g^{(\star)}_{\text{tissue}}(x^{(\star)}_{\text{tissue}};u,v,\theta) &= g^{(r)}_{\text{tissue}}(x^{(r)}_{\text{tissue}};u,v,\theta) \\ &\quad+\sum_{u,v,r_\theta}\left[\delta_{((u,v,\theta),(h,j,k)}\left[x^{(\star)}_{(h,j,k)}-x^{(r)}_{(h,j,k)}\right]\right]. \end{aligned} \tag{6.14}$$

We can therefore also usefully store two arrays indexed by (u, v, r_θ), where projection r_θ is acquired at angle θ. One of these contains the measured pixel values $y_{(u,v,\theta)}$. The other contains the current values of $g_{\text{tissue}}(x; u, v, \theta)$, which are initially generated using the ray-trace, and subsequently updated whenever a proposed voxel value is accepted using Equation 6.14.

Storing these arrays in RAM allows us to run the sampler reasonably quickly. However, they are large structures, which limits the size of the reconstruction task which can be tackled. For example, if we use 600 projections, and assume that each voxel is associated with exactly one pixel in each of these projections, then we need to store 600 sets of indices (u, v, r_θ) and path lengths $\delta_{(\cdot)}$. If each of the indices is stored as a 16-bit integer, and the path length as a 32-bit single-precision float, then each voxel in the reconstruction requires at least 600×10 bytes (nearly 6kB) of RAM. If we were to attempt to include the whole of the patient's pelvis in the sampler at a resolution of 1mm, we would need approximately $416 \times 416 \times 144$ voxels, requiring nearly 140GB of RAM.

This is obviously not possible on a desktop computer (where 8GB of RAM is typical), especially as the above discussion assumes that each voxel is associated with only one pixel per projection—in fact, many voxels are on the path to two or more pixels—and has not included the requirement to store the measured data or the $g(\cdot)$ values. We therefore need to restrict the task to one for which we can hold the necessary data in RAM. We can do this in two ways: by reducing the number of projections we use, and by reducing the number of voxels which we update.

Reducing the number of projections is trivial. We shall present results with differing numbers of projections. These demonstrate that it is not necessary to use all 600 projections to obtain a good reconstruction—around 60 projections is sufficient. The reduction possible whilst maintaining the quality of the reconstruction is not enough to enable storage of the required data in RAM.

Reducing the number of voxels is not trivial. We define a volume of interest (VOI) $\mathcal{V}$ within the larger reconstructed volume $\mathcal{R}$, such that we can store all the necessary data in RAM. This VOI is typically tens of voxels in each of the three dimensions. We can update the voxels in $\mathcal{V}$ many times, sampling from the conditional distribution of the attenuation in $\mathcal{V}$ given the values in the rest of the reconstruction (i.e. in $\mathcal{R} \setminus \mathcal{V}$) as well as the data.

We could move this VOI round (for example, using a semi-overlapping scheme) and update all of the tissue attenuation values, one VOI at a time. After the VOI has made several passes over the whole volume, we would expect to obtain samples from the posterior distribution $\pi(X_{\text{tissue}}|y)$.

However, we are only really interested in knowing the tissue attenuation in the region around the markers. We therefore use a different method to estimate the tissue attenuation values in $\mathcal{R} \setminus \mathcal{V}$: the Feldkamp-Davis-Kress (FDK) reconstruction. We then sample the posterior distribution of the values in $\mathcal{V}$, given these values in $\mathcal{R} \setminus \mathcal{V}$.

Note that our model for those voxels in $\mathcal{R} \setminus \mathcal{V}$ has not changed—it is only in our analysis of the the posterior density that we have imposed the distinction between $\mathcal{V}$ and $\mathcal{R} \setminus \mathcal{V}$. The model is defined for all of $\mathcal{R}$. We initially assign values to all voxels by using an FDK reconstruction. In our sampling of the posterior distribution, we update the values of only the voxels in $\mathcal{V}$.

At the outer surface of $\mathcal{V}$, we need to consider the prior probability of voxel values. In choosing only to update a VOI, we are defining a subset of the sample space in which we are happy to restrict our sampling. The calculation of the prior probability of a pixel value at the edge of $\mathcal{V}$ is the same as for any other voxel: we consider the six nearest neighbours, some of which will be in $\mathcal{V}$ and some not. We shall examine how this affects the voxel reconstruction.

An alternative to storing the arrays in RAM, with the resultant necessity to restrict the reconstructed volume, would be to calculate the values of $\delta_{(\cdot)}$ at every iteration. For each voxel, this would require determining a set of pixels which will include all those for which $\delta_{(\cdot)} \neq 0$, then ray tracing from the X-ray source to the centre of each of these pixels. Doing this for hundreds of projections for each voxel at each iteration would be prohibitively expensive; even reducing the number of projections leaves an extremely daunting task.

The proposal distribution for the tissue attenuation values was normal, with the mean being the current value and variance 1000^2 (chosen so that the acceptance probability was approximately 30%).

Updating the marker parameters

In order to calculate the contribution to the likelihood from a particular pixel value in one projection, we need to know the volume of marker which is contained in the pyramid with the pixel at its base and the X-ray source at its apex. For the straight markers, it would be reasonably straightforward to calculate this exactly. However, the analysis presented here is designed to be easily extendable to other shapes of marker, where this may not be so easy to calculate.

We used the same stratified sampling approach as in Section 6.3.3. For each marker, we sample a point in each of the $N = 1000$ discs whose central points are evenly spaced along the central axis of the cylinder. Within each disc, the point is sampled uniformly by area.

Each sampled point was then projected using a transformation matrix pre-calculated for each projection. The number of points projected onto each pixel then gives an estimate of the volume of marker contained within the relevant pyramid.

The joint prior for the two markers requires knowledge of whether they intersect—if they do, the prior probability is zero. An estimate of $\mathbb{I}_{\text{disjoint}}(x_{\text{marker}})$ (the indicator of whether the markers intersect or not) is given by whether points from both markers are generated in any voxel.

To update the markers, one of the twelve scalar parameters (three per marker end) was chosen at random. The proposal was again normally distributed with the mean being the current value, and variance 0.01^2, again chosen so that the acceptance rate was approximately 30%.

6.5 Results

6.5.1 Overview

In this section, we shall present the results from Markov chain Monte Carlo (MCMC) simulations using various phantoms and patient data. We shall demonstrate the performance of the model for the tissue attenuation alone (updating only X_{tissue}), the markers alone (updating only X_{marker}), and both tissue attenuation and markers together (updating both X_{tissue} and X_{marker}).

Phantom experiments

We shall first present results using phantom data. We shall show good results using the tissue attenuation model alone to create reconstructions of phantoms with and without markers, with fewer artefacts than when we use an FDK reconstruction. When there are markers in the phantom, they appear in the reconstruction as voxels with very high values for the tissue attenuation.

In the artificial situation of a phantom containing markers, we can examine the performance of the marker model alone by fixing the tissue attenuation values (and hence the values of $g_{\text{tissue}}(x; u, v, \theta)$) to be those generated by an FDK reconstruction of the same phantom without markers. Using only the marker model, we find we can accurately locate the markers with plausible credible intervals which contain the true values of the marker end coordinates.

When we include both the tissue attenuation X_{tissue} and the marker parameters X_{marker} in the model, we find that the performance depends on the value of β (the tuning parameter in the prior for the tissue attenuation values). For low values of β, our sampler becomes stuck in a local mode and does not accurately locate the markers. For high values of β, we obtain accurate estimates of the marker location, but do not recover any of the structure in the tissue attenuation.

Application to patient data

We shall also present results using patient data. As for the phantoms, including only the tissue attenuation in the model gives good reconstructions of the patient, with fewer artefacts than in the FDK reconstructions. This is the case in regions with and without markers. Where there are markers, we see them as voxels with very high tissue attenuation values.

With real patient data, we do not know what the tissue attenuation values would be in the absence of the markers. We cannot therefore isolate the marker process in the same way as for the phantom.

From the phantom experiments, we already know that we are unlikely to be able to reconstruct the tissue attenuation at the same time as accurately estimating the locations of the markers. We shall present the results from analysing the posterior distribution with both the tissue attenuation X_{tissue} and the marker parameters

X_{marker} included in the model, with a high value of β (so that we do not expect to recover structure in the tissue attenuation values). We find that artefacts in the FDK reconstruction cause the estimates of the marker locations to be inaccurate.

Although we have not been able to find the tissue attenuation values at the same time as the marker locations, we have found a different way to include only the marker parameters X_{marker} in the model. This has allowed us to create accurate estimates of the marker location (which is our primary goal), the details of which we shall describe later.

6.5.2 Modelling the tissue attenuation only (phantom data)

In this section, we shall show reconstructions of phantoms generated by sampling from the posterior distribution of the tissue attenuation values given the data, $\pi(X_{\text{tissue}}|y)$. We shall use three phantoms: a very small phantom (which does not contain markers), and a large phantom both with and without markers (although we will not, in this section, explicitly model the markers).

The phantoms used are variations of the Shepp-Logan phantom. This is a 2D mathematical phantom designed to simulate a brain, with ellipses representing various structures [34]. The phantoms in this section were generated using Matlab, which includes a built-in function, `phantom`, for this purpose.

Very small phantom with no markers

We created a phantom of size $64 \times 64 \times 8$ voxels, with isotropic pitch of 1mm. The four central 64×64 slices consisted of the 2D Shepp-Logan phantom. One of these slices in illustrated in Fig. 6.5. The other four slices contained no attenuation.

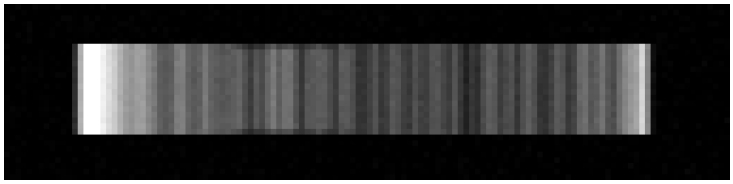
We created 616 projection images by ray-tracing, at angles equally spaced between 0 and 2π, and with no detector offset. The geometry was otherwise the same as for the patient scans. Additive Gaussian noise was incorporated, with the addition to each pixel being independent. Fig. 6.6 shows one of the projections with no added noise, and added noise with variance 0.1^2 and 1^2.

Fig. 6.7 shows a central slice from the FDK reconstructions obtained by using each of these three data sets. Note that no smoothing filter was used in these reconstructions; a better result might be obtained if one were used.

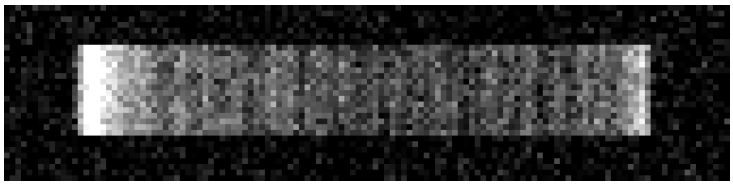
Figure 6.5: The 64 × 64 pixel Shepp-Logan phantom.

The 2D Shepp-Logan phantom of size 64 × 64 pixels. For display purposes the greyscale ranges from 0 (black) to 0.1 (white); this range encompasses all structures except for the 'skull' (white ellipse), which has a pixel value of 1.

Before adding noise



Added noise with variance 0.1^2.

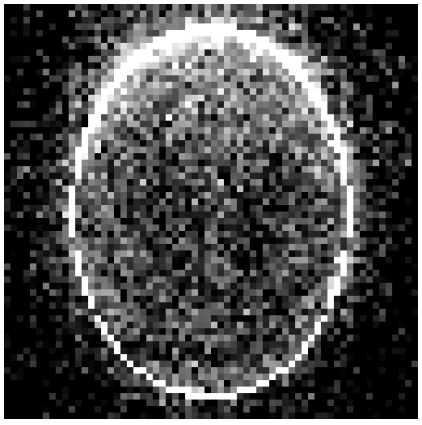


Added noise with variance 1^2.

Figure 6.6: Projections of the $64 \times 64 \times 8$ phantom.

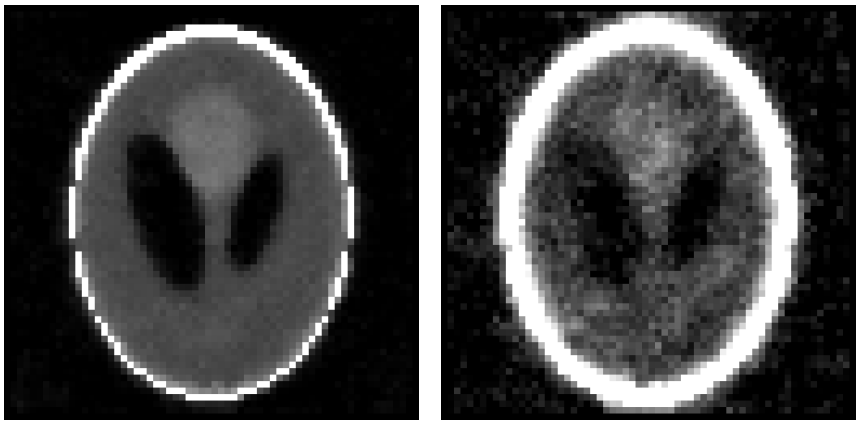
Noise-free data.

Data with added noise (variance 0.1^2).



Data with added noise (variance 1^2).

Figure 6.7: FDK reconstructions generated from each of the three data sets.



Added noise with variance 0.1^2. Added noise with variance 1^2.

Figure 6.8: Central slices from the MPM reconstruction.

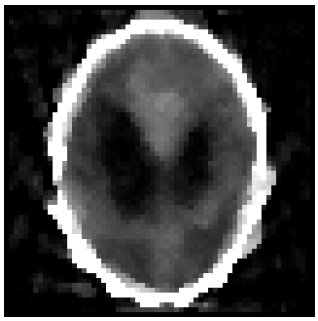
Fig. 6.8 shows central slices from the marginal posterior mean (MPM) reconstruction obtained using 62 (i.e. one in ten) of the projection images, for both the data sets with added noise. In each case, the MCMC sampler was initialised with an array of zeros. The chain was run for 4000 iterations, and the first 500 discarded as burn-in. Even though many fewer projections are used, the reconstructions are better than those obtained using the FDK algorithm.

Fig. 6.9 shows central slices from the MPM reconstruction obtained using differing numbers of projection images, using the data with added noise of variance 0.1^2. These demonstrate how the quality of the reconstruction increases with increasing numbers of projection images. Only minor improvements are seen after 62 projections; this is the number that was chosen for the rest of the simulations whose results are shown here.

Large phantom with no markers

A larger phantom was generated, again using Matlab, with each of 144 slices consisting of a 2D Shepp-Logan phantom of size 256×256 pixels. One of these slices is shown in Fig. 6.10. This phantom is too large to be able to hold the necessary data in RAM to be able to sample the posterior distribution for the whole phantom as we did for the very small phantom. Instead, the sampling was restricted to a VOI

8 projections.



16 projections.

31 projections.

62 projections.

124 projections.

308 projections.

616 projections.

Figure 6.9: The effect of varying the number of projections.

Central slices from the MPM reconstruction with varying number of projections used.

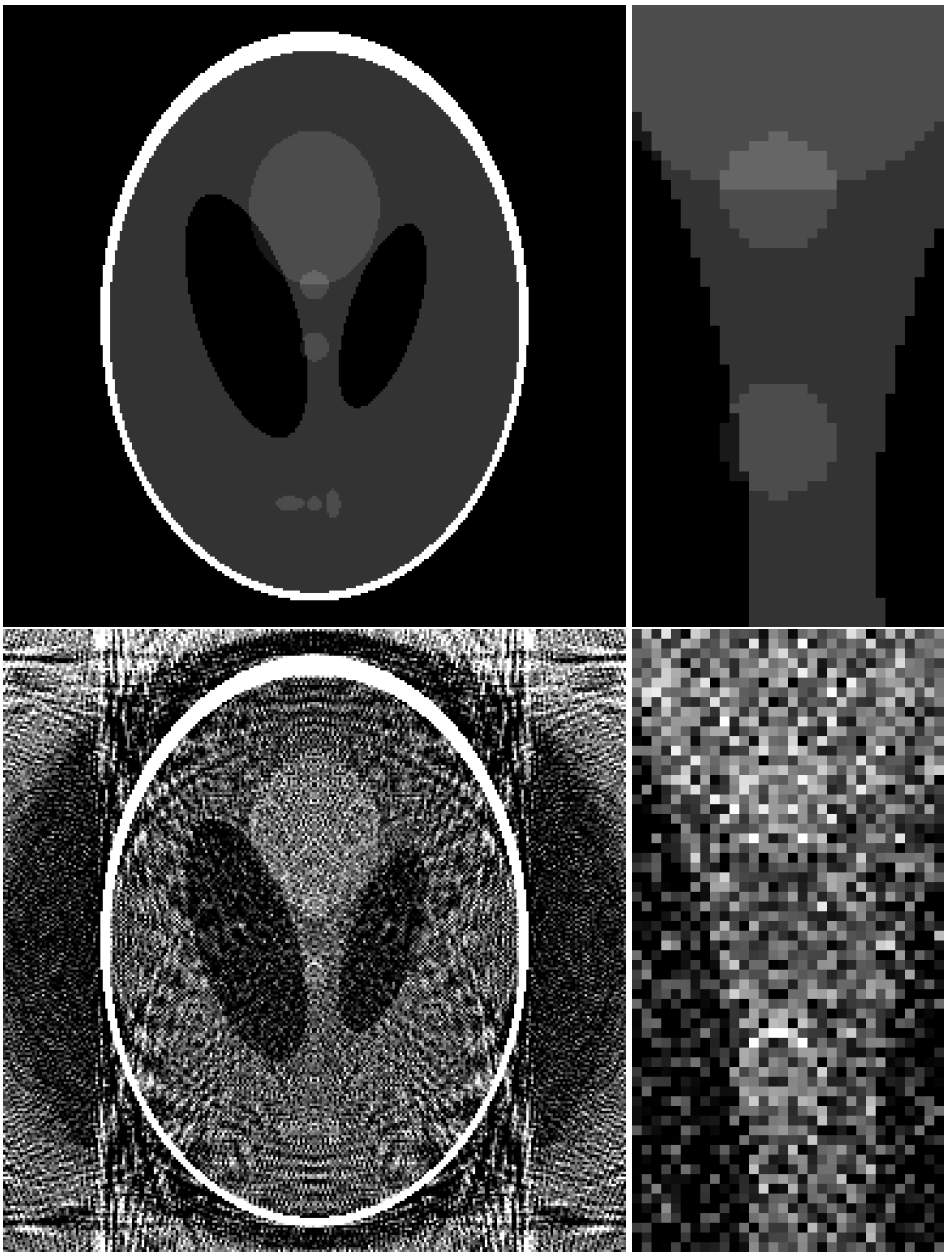
of size 32×64 pixels in the transaxial plane, and 12 voxels in the direction parallel to the axis of rotation. This is also shown in Fig. 6.10.

Ray-tracing was again used to obtain 616 projections of this volume, to which additive Gaussian noise with standard deviation 0.1^2 was added. These projections were used to reconstruct the volume using the FDK algorithm, which is also shown in Fig. 6.10.

Fig. 6.11 shows the MPM reconstructions obtained by sampling the posterior distribution of the voxel values in the VOI, given the noisy projection data. In each case, the data from 62 projections were used. We need to assume values for the voxels in the volume outside the VOI. For the case of the phantom, we know what the voxel values are; the MPM values obtained in the VOI using this knowledge of the rest of the phantom are shown in the top row of Fig. 6.11.

Of course, when using real data, we do not know the true voxel values in the volume outside the VOI. Instead, we need to use values generated from a less memory-intensive reconstruction, such as FDK. The bottom row of Fig. 6.11 shows the MPM values obtained when the FDK reconstruction was used for the voxel values outside the VOI. The FDK reconstruction contains artefacts, which result in a much poorer reconstruction in the VOI.

This reconstruction is qualitatively different to the FDK reconstruction, and not obviously better. However, we note that the purpose of finding the tissue attenuation values is not to compete with the FDK reconstruction in marker-free regions—indeed, it would not be a fair comparison, as the FDK reconstruction we are using is very simple and does not include a noise-reducing filter. The MPM reconstruction in the VOI captures the gross features of the phantom, such as the low voxel value (black) areas, which is what we shall need it to do in order to model the markers as a separate random process.

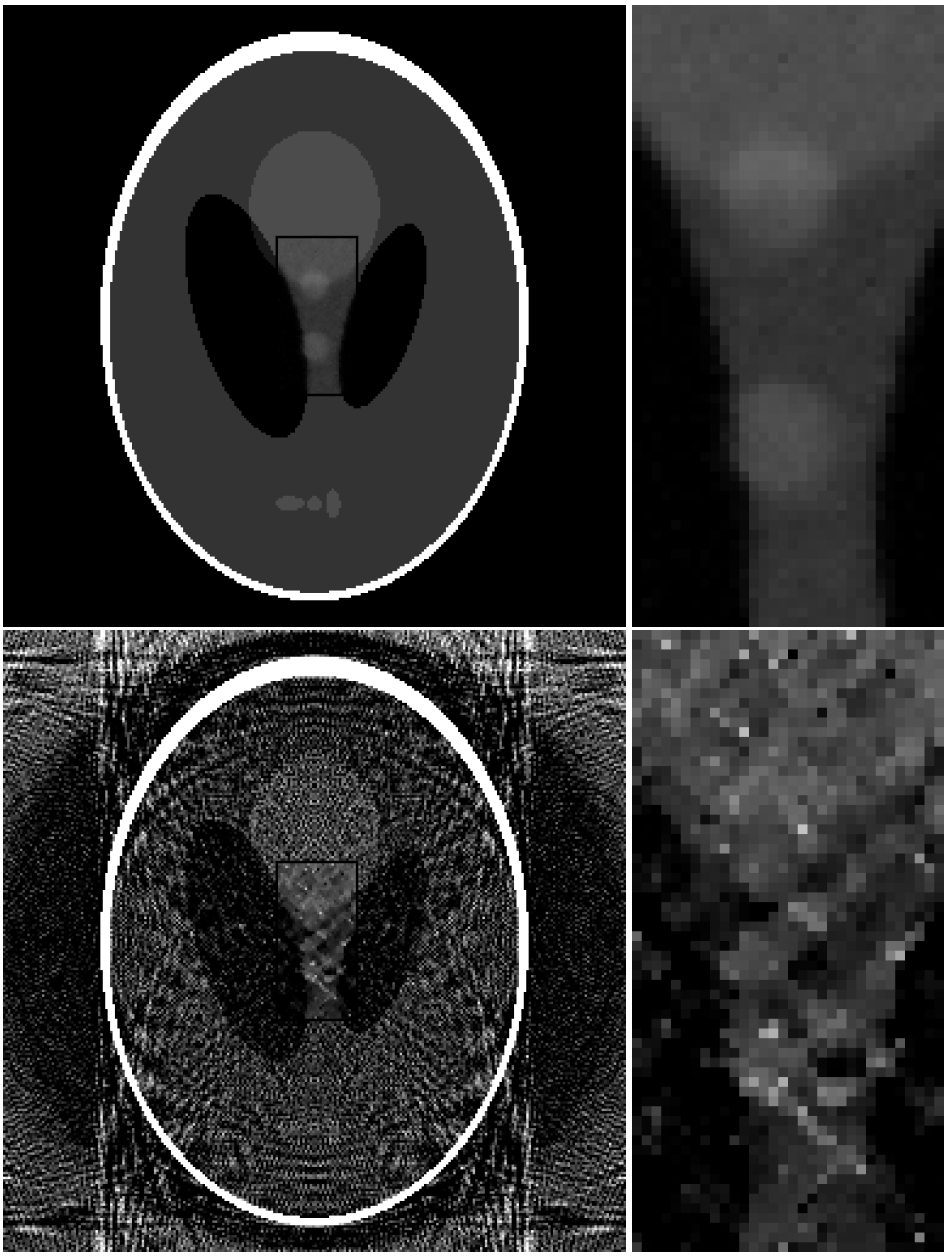


Transaxial slices through the centre of the phantom.

The VOI only.

Figure 6.10: The large phantom and FDK reconstruction.

Top: The 256 × 256 Shepp-Logan phantom. For display purposes the greyscale ranges from 0 (black) to 0.1 (white); this range encompasses all structures except for the 'skull', which has a pixel value of 1. Bottom: The FDK reconstruction generated from noisy projections.



Central slices, showing the VOI (outlined in black) and surrounding volume.

The VOI only.

Figure 6.11: MPM reconstructions of a central VOI.

Top: VOI embedded in the truth (the Shepp-Logan phantom). Bottom: VOI embedded in an FDK reconstruction generated from noisy projections.

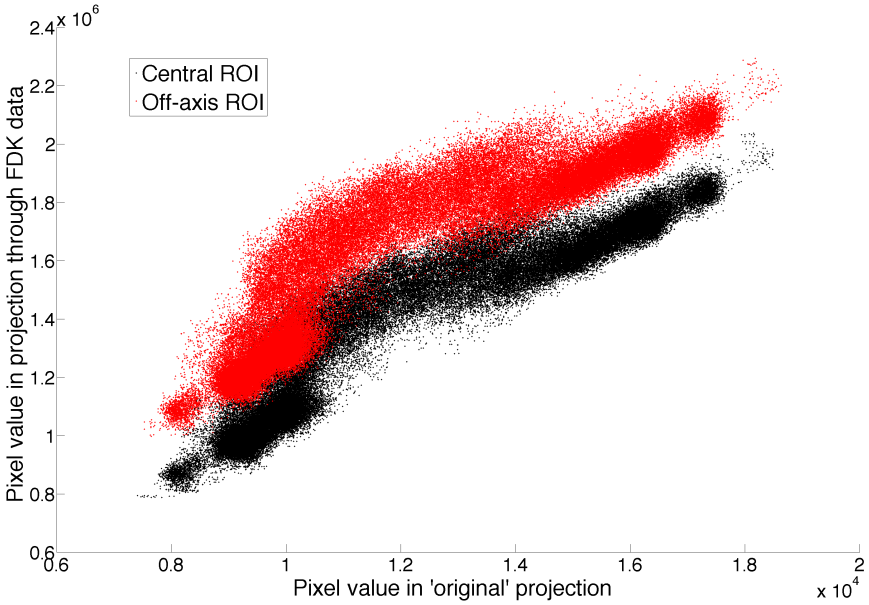


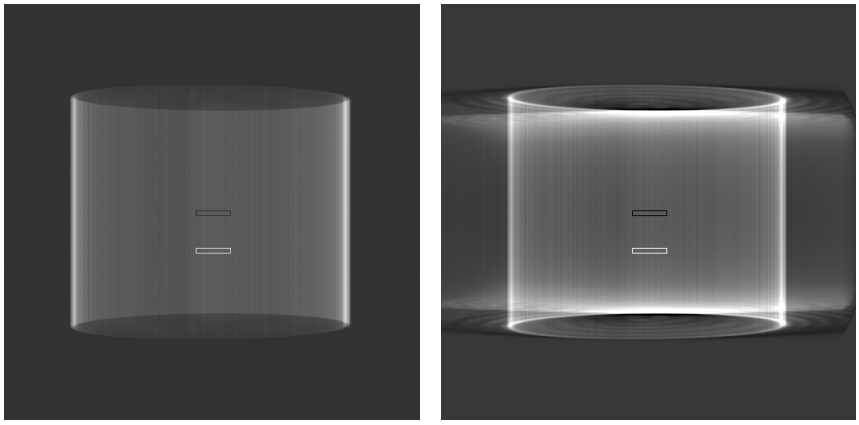
Figure 6.12: Pixel values in 'rescaling' ROIs.

Scatter plot of the pixel value in the 'original' projections and in the projections through the FDK reconstruction.

Rescaling the reconstruction

To calculate the marginal likelihood of a particular measured pixel value given a configuration of voxel values, we need to know the integral of voxel values along the path from the X-ray source to that pixel. However, the FDK reconstruction is relative—the scale is lost in the reconstruction—and the projection values are scaled and centred (using unknown values) to become the pixel values Y. We therefore scale and offset the voxel values so that the calculated projected pixel values match the measured values.

We do this by creating projections of the FDK reconstruction, with geometry matching the measured projection data, and compare pixel-by-pixel. We calculate the necessary scaling and offset values by carrying out a least-squares fit using the reconstructed value and the path-length through the reconstruction, from a region of interest (ROI) in each projection. The change in voxel value due to the rescaling was found to be much greater than that due to the offset.



The 'original' image.

The projection through the FDK reconstruction.

Figure 6.13: ROIs used in rescaling the FDK reconstruction.

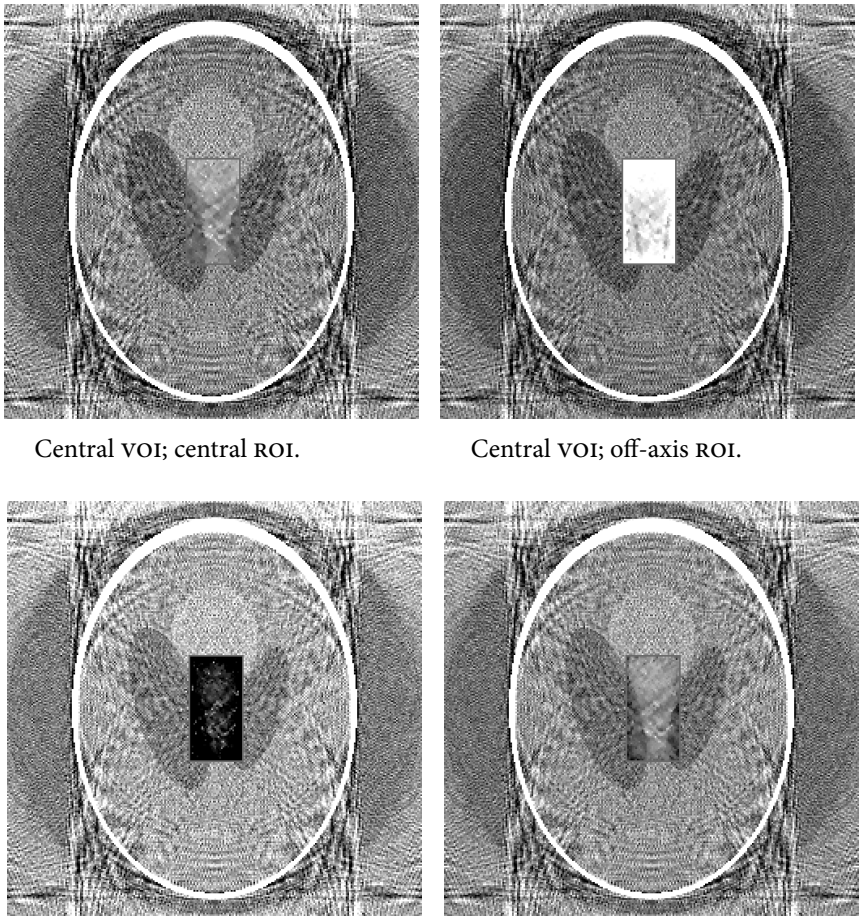
The ROIs are illustrated for one projection of the Shepp-Logan phantom. In each image, the central ROI is outlined in black and the off-axis ROI outlined in white.

However, the reconstruction process is inexact, and corrupted by reconstruction artefacts, and artefacts due to noise in the projection data. (In real data, there are additional artefacts due to the energy dependence of the attenuation and the presence of scattered radiation.) Fig. 6.12 shows the relationship between the pixel value in the 'original' data (that is, the synthetic data with added Gaussian noise), and the projections obtained through the FDK reconstruction. There are two data sets shown on the scatter plot: in black, the pixel values from an ROI in the centre of each of the images; in red, those from an ROI which is offset from the central axis. These ROIs are shown in Fig. 6.13.

Fig. 6.14 shows the effect of changing the ROI used to determine the rescaling. The MPM reconstructions which best match the truth are those where the rescaling ROI is at the position in the image where the data used for the likelihood calculation are located. We shall return to the issue of choosing a rescaling ROI in Section 6.5.2.

Large phantom containing markers

A phantom containing markers was generated, again based on the Shepp-Logan phantom of size 256×256, replicated in 144 slices. Each of the voxels in this



Central VOI; central ROI. Central VOI; off-axis ROI.

Off-axis VOI; central ROI. Off-axis VOI; off-axis ROI.

Figure 6.14: The effect of changing the location of the rescaling ROI.

Each image is a central slice showing the MPM reconstruction of a central VOI embedded in a larger reconstruction. The ROI is either at the centre of each of the projection images, or off axis, at the level where the centre of the off-axis VOI is projected.

phantom has a value between 0 (representing air) and 1 (representing skull), as illustrated in Fig. 6.10.

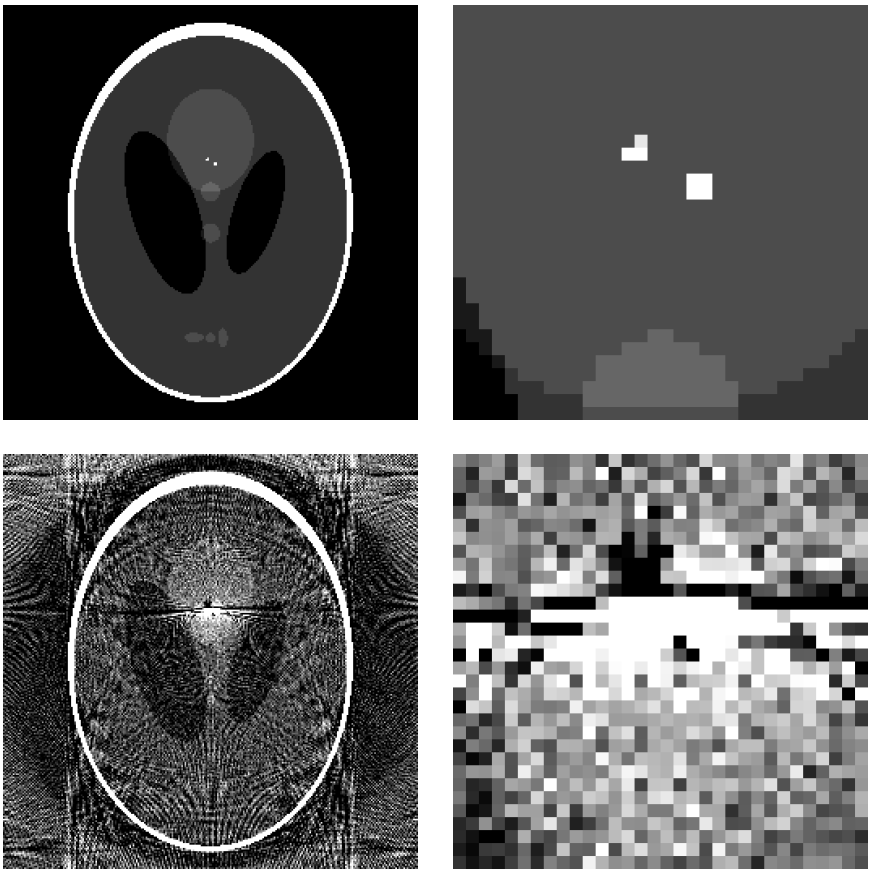
To emulate markers in the phantom, we changed some of the voxel values in this phantom. We defined two end points per marker. One of the markers was created with a length of 10mm, and one with a length of 20mm. We then generated $N = 1,000,000$ points in each marker using the stratified sampling approach described in Section 6.3.3. For each of the $2N$ points, the voxel value for the voxel containing that point was increased by $0.04/l$, where l is the length of the relevant marker in millimetres. Projection data were created by ray-tracing, and adding Gaussian noise with variance 0.1^2.

A slice from this phantom is illustrated in Fig. 6.15. Also shown is the FDK reconstruction, in which the markers cause serious star artefacts. Note that, in these synthetic data, we have not included any modelling of the energy dependence of the attenuation. The artefacts in a real data might therefore be expected to be worse. (As previously noted, this is a rudimentary FDK reconstruction, and including a noise-reducing filter might improve the reconstruction.)

Fig. 6.16 shows the MPM reconstruction obtained in a VOI in two scenarios: with the VOI embedded in the truth, and with the VOI embedded in an FDK reconstruction obtained using the noisy projection data. The reconstruction with the VOI embedded in the truth is very good. The MPM reconstruction in the VOI embedded in the FDK reconstruction is much better than the FDK reconstruction in this VOI (Fig. 6.15), with the star artefacts completely removed.

Rescaling the reconstruction

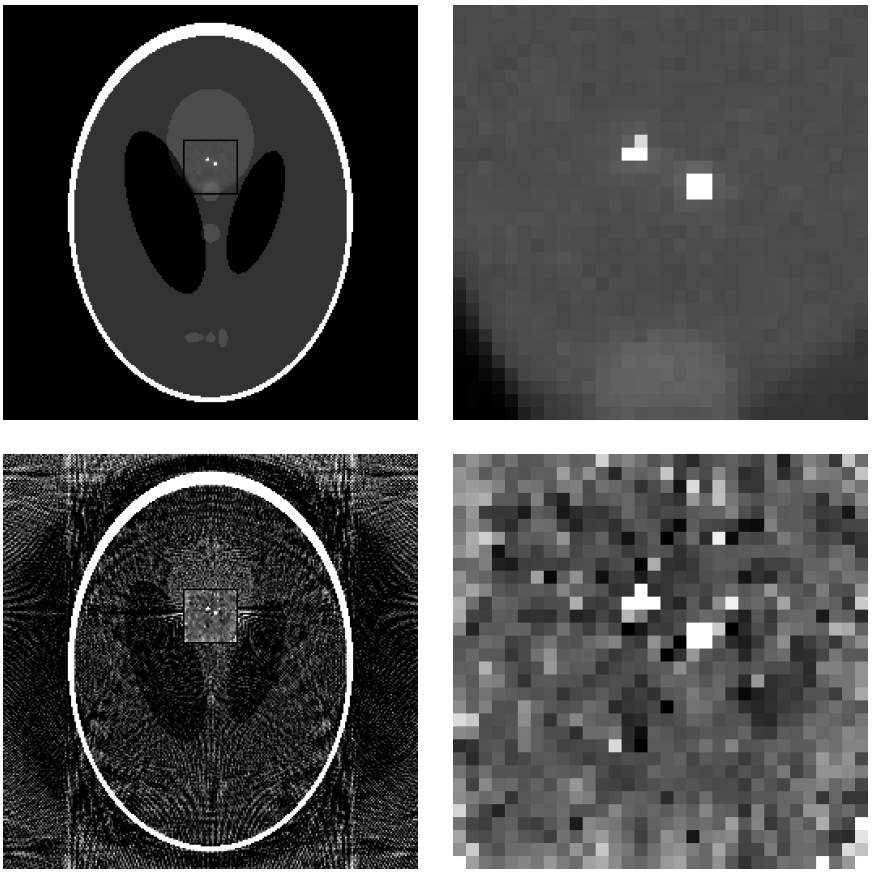
In Section 6.5.2, we discussed the rescaling of the FDK reconstruction (so that the values obtained by ray-tracing through the reconstruction match the measured pixel values). We concluded that MPM reconstructions which were closer to the truth when the ROI used to estimate the rescaling factors was close to the point in the image where the VOI is projected. That is, if the VOI was on the central axis, rescaling ROIs on the central axis gave the best results; if the VOI was off-axis, it was better to use off-axis ROIs.



Central slices. The VOI to be sampled only.

Figure 6.15: Slices from the phantom with markers.

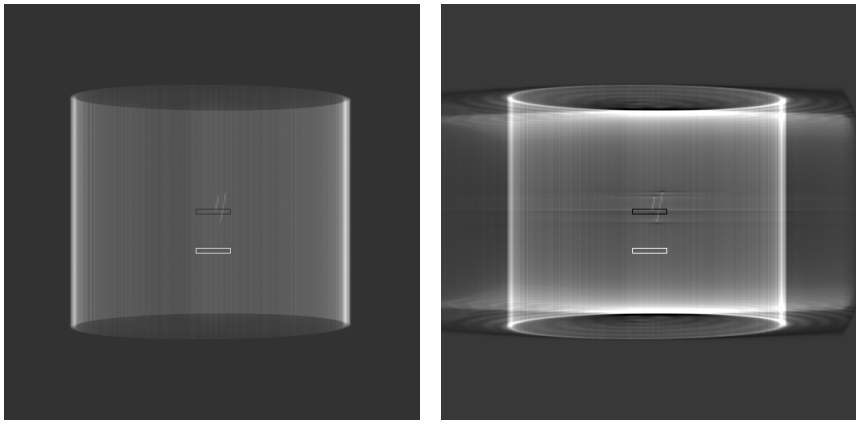
Top: Original phantom. Bottom: FDK reconstruction generated from noisy projections.



Central slices. The VOI only.

Figure 6.16: MPM reconstructions of phantom with markers.

MPM reconstruction of a VOI embedded in a larger reconstruction. Top: VOI embedded in the truth (the Shepp-Logan phantom). Bottom: VOI embedded in an FDK reconstruction generated from noisy projections.



The 'original' image.

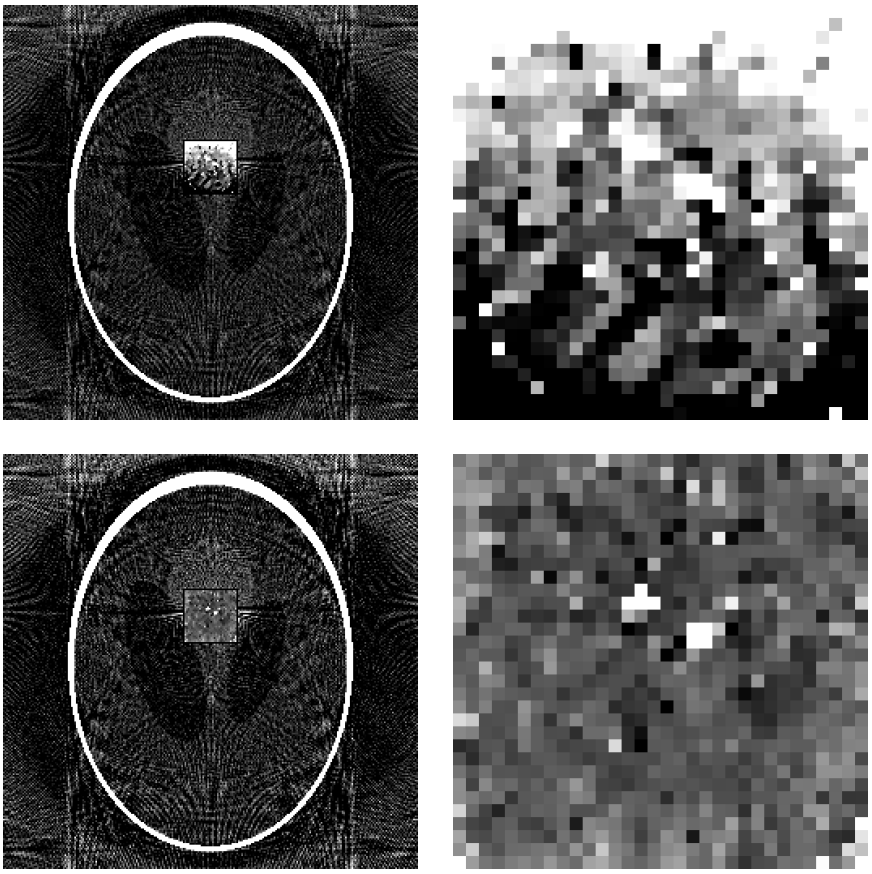
The projection through the FDK reconstruction.

Figure 6.17: ROIs used in rescaling the FDK reconstruction.

The ROIs are illustrated for one projection of the phantom containing markers (c.f. Fig. 6.13, which shows the same ROIs in the phantom without markers). In each image, the central ROI is outlined in black and the off-axis ROI outlined in white.

In the case where there are markers present, there are additional artefacts in the reconstruction. When rescaling ROIs are defined in the projections through this reconstruction, the artefacts also appear in the projections, as shown in Fig. 6.17. These can cause problems with the rescaling, as shown in Fig. 6.18. The artefacts are projected into the central ROIs. This means that using off-axis ROIs gives an MPM reconstruction closer to the truth than using central ROIs, even when the VOI is on the central axis.

This illustrates the necessity of choosing a set of pixels on which to base the estimation of the rescaling factors which are uncorrupted by artefacts, or at least as near to this as possible, and which are close to the the area of the projection image where the VOI is projected. This creates a tension, as the VOI usually contains the markers, which are the source of some of the worst artefacts in the FDK reconstruction.



Central slices. The VOI only.

Figure 6.18: The effect of changing the rescaling ROI, with markers.

MPM reconstructions of a central VOI embedded in a larger reconstruction. Top: ROI used for rescaling at the centre of each projection image (at the location where the VOI is projected). Bottom: ROI used for rescaling located away from the central axis.

6.5.3 Modelling the marker attenuation only (phantom data)

Large phantom containing markers

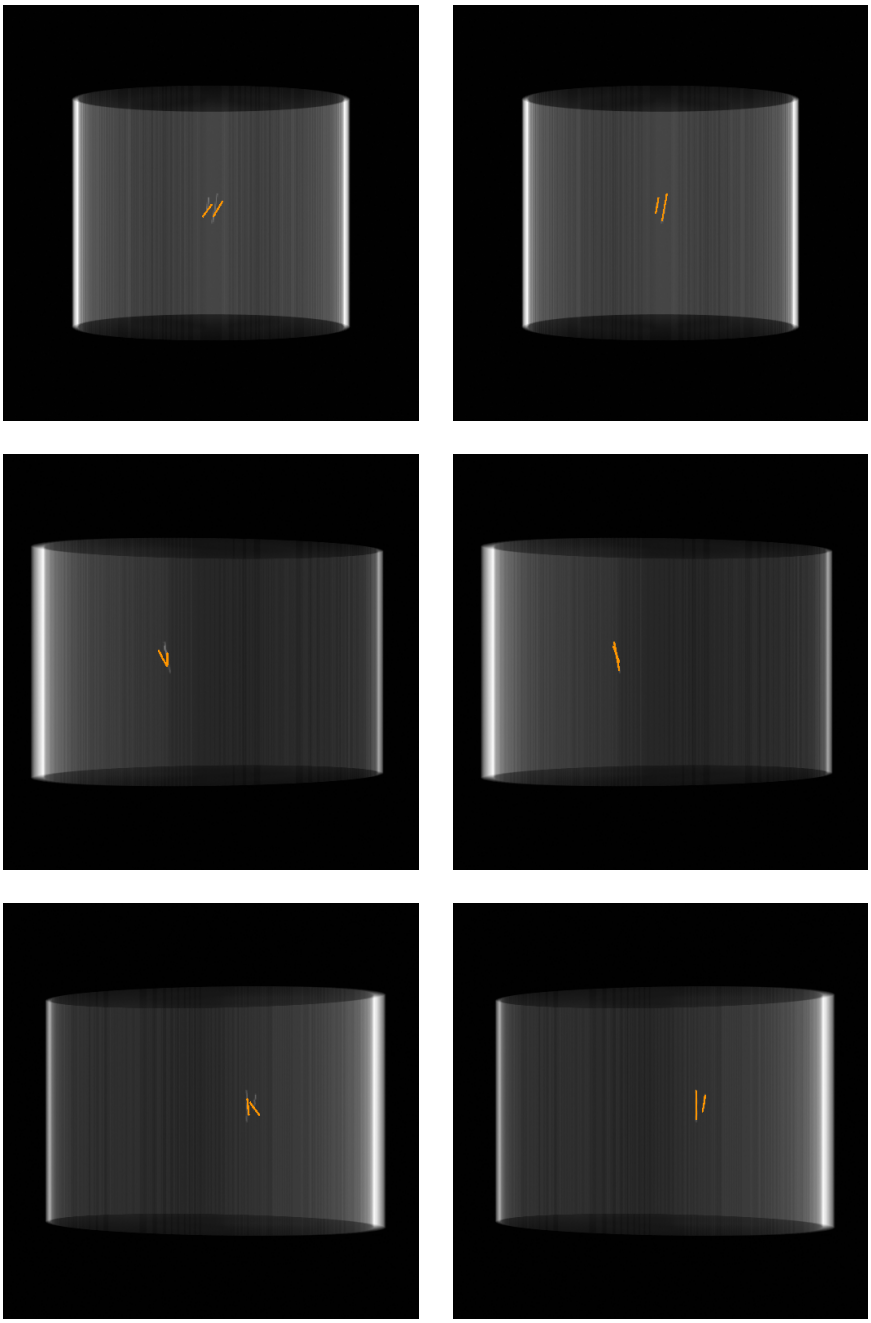
The results presented so far have shown that reasonable estimates of the tissue attenuation values can be obtained by using part of our model. In this section, we shall demonstrate the performance of the other part of our model, focussing on estimating the location of the markers.

We shall use the same phantom as in Section 6.5.2, that is the large phantom containing markers. In that section, we generated projection data by ray tracing through this phantom and adding Gaussian noise; this will be our measured data y.

In this section, we want to include only the marker parameters X_{marker} (and not the tissue attenuation X_{tissue}) as random variables. However, we need to include known values of the tissue attenuation (in the absence of markers) in the calculation of the likelihood. By using phantom data, we have created the artificial situation where we conveniently have a source of such information—the FDK reconstruction of the large phantom without markers (as used in Section 6.5.2). That is, we calculate the values of $g(u, v, \theta; x_{\text{tissue}})$ by performing a ray trace on the FDK reconstruction obtained using the noisy projections of the phantom with no markers. The values of $g(u, v, \theta; x_{\text{tissue}})$ are no longer functions of random variables, but functions of the tissue attenuation which is assumed known.

The posterior distribution of the position of the marker ends was sampled using the Metropolis-Hastings sampler, with 500,000 iterations. Recall that we parameterise the markers using the four end points, each of which is a location in $\mathbb{R}^3$. Six projections of the initial location of the markers are shown in Fig. 6.19, along with a sample from the posterior (in fact, iteration 200,000), each superimposed on the 'original' data. Figures 6.20-6.23 show the sample paths of the twelve coordinates; Figures 6.24-6.27 show histograms of the samples (after discarding the first 100,000 samples as burn-in).

Table 6.1 gives the true values for each of the coordinates, with the marginal posterior mean. Also shown are 95% credible intervals. These were estimated as the values L and U such that 95% of the samples were in the interval $[L, U]$, and



Initial position.

Sample from the posterior distribution.

Figure 6.19: Projections showing the location of the markers.

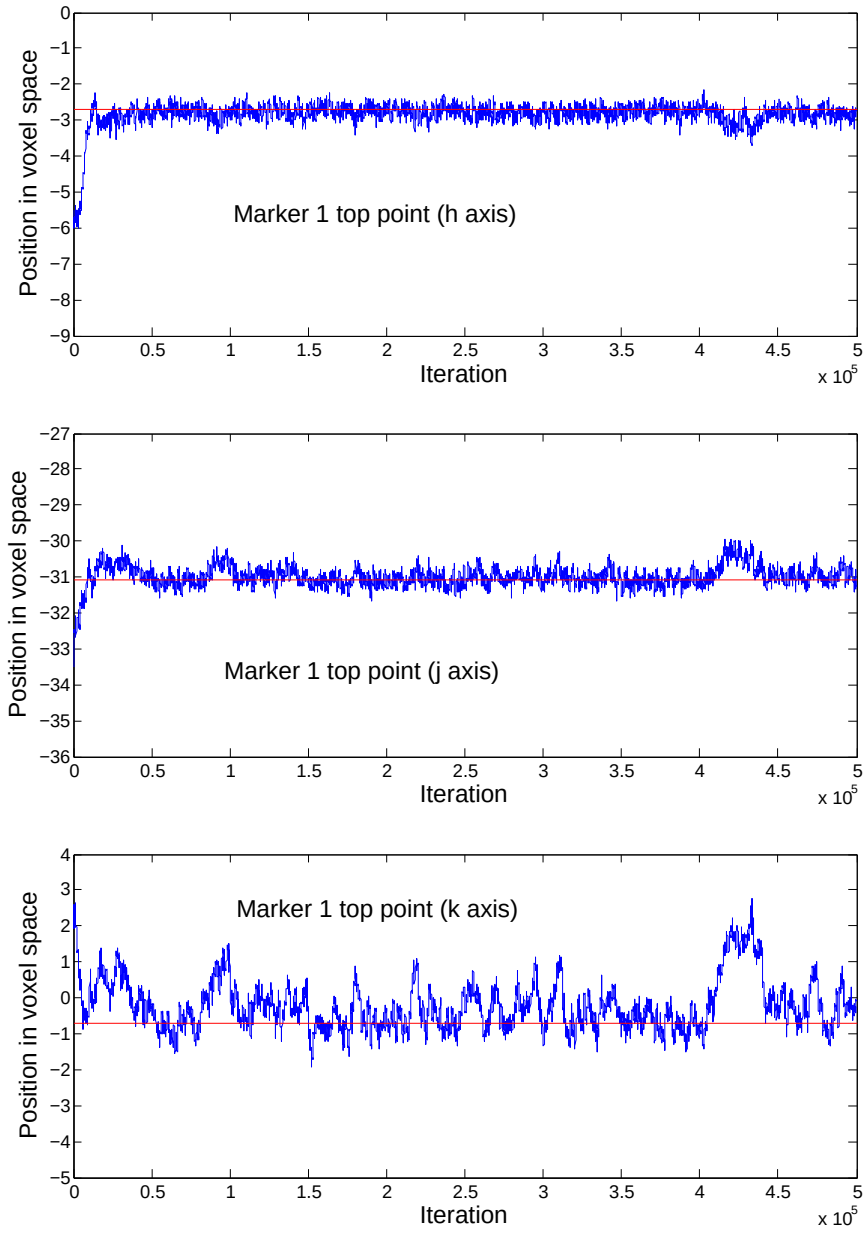


Figure 6.20: The sample path for marker 1 (top point).

The sample paths of each of the three coordinates in the MCMC simulation (blue). The true value is also shown (red).

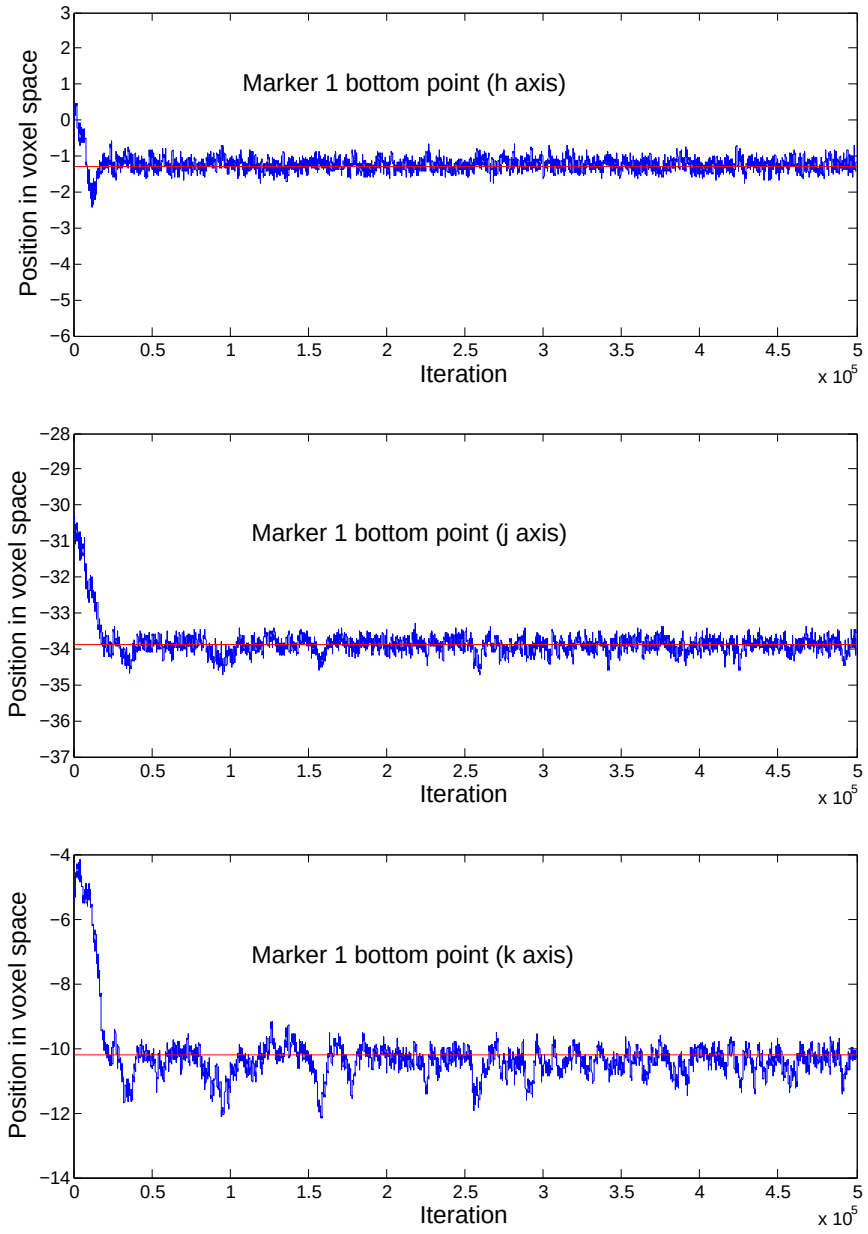


Figure 6.21: The sample path for marker 1 (bottom point).

The sample paths of each of the three coordinates in the MCMC simulation (blue). The true value is also shown (red).

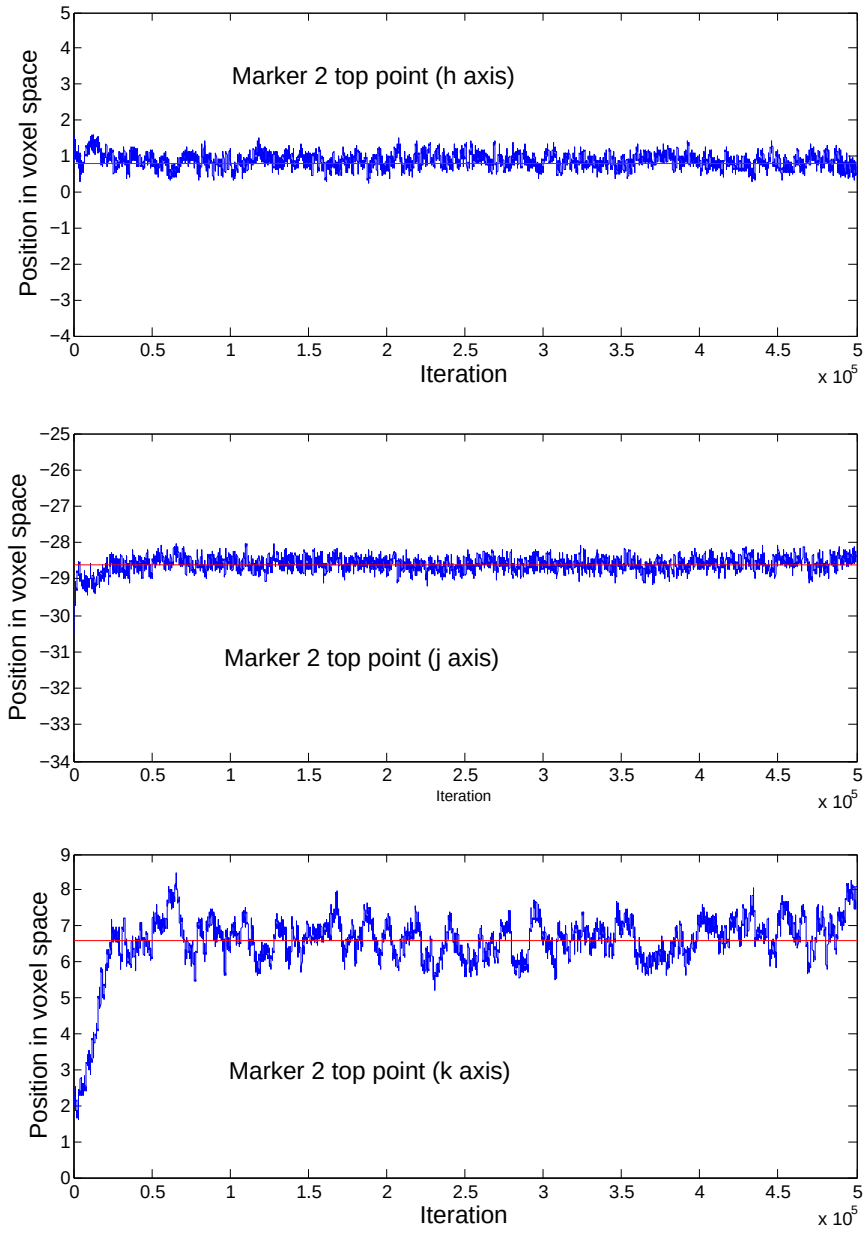


Figure 6.22: The sample path for marker 2 (top point).

The sample paths of each of the three coordinates in the MCMC simulation (blue). The true value is also shown (red).

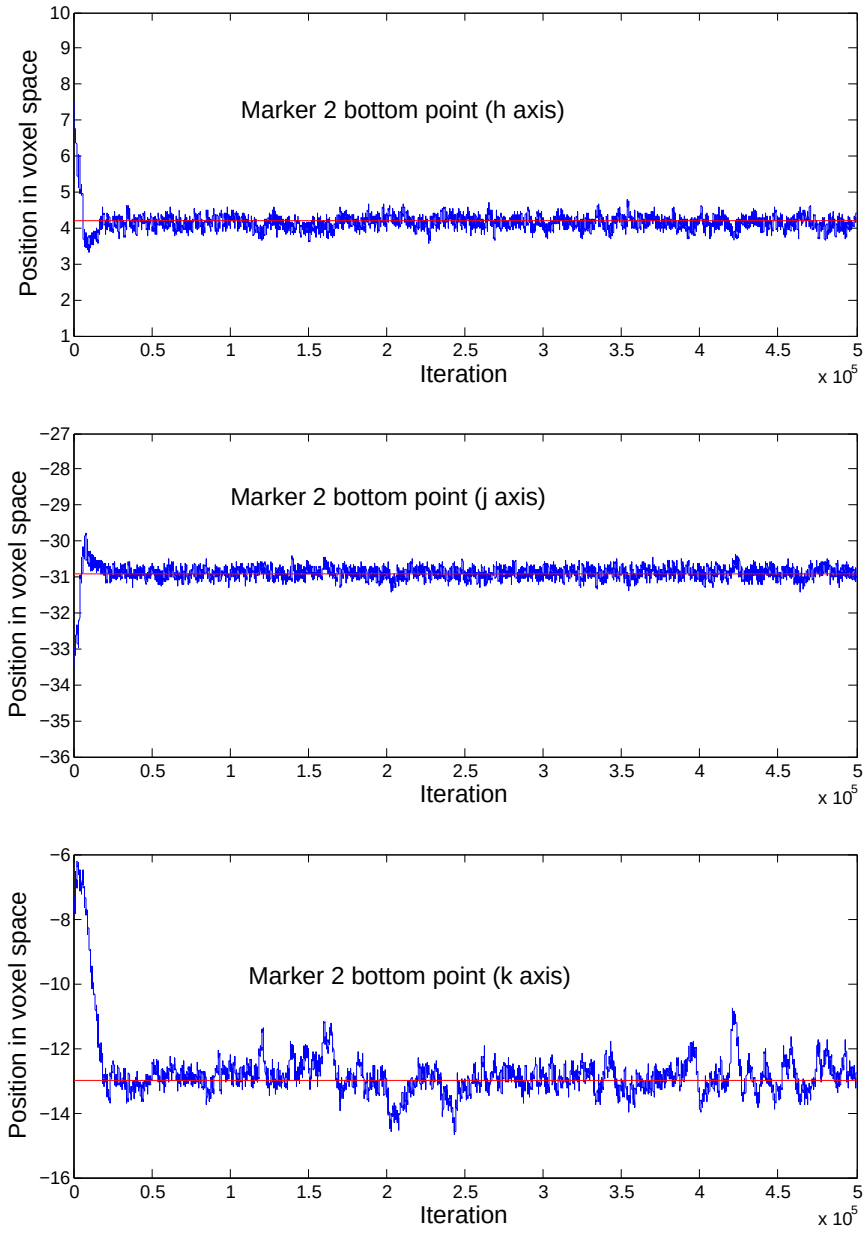


Figure 6.23: The sample path for marker 2 (bottom point).

The sample paths of each of the three coordinates in the MCMC simulation (blue). The true value is also shown (red).

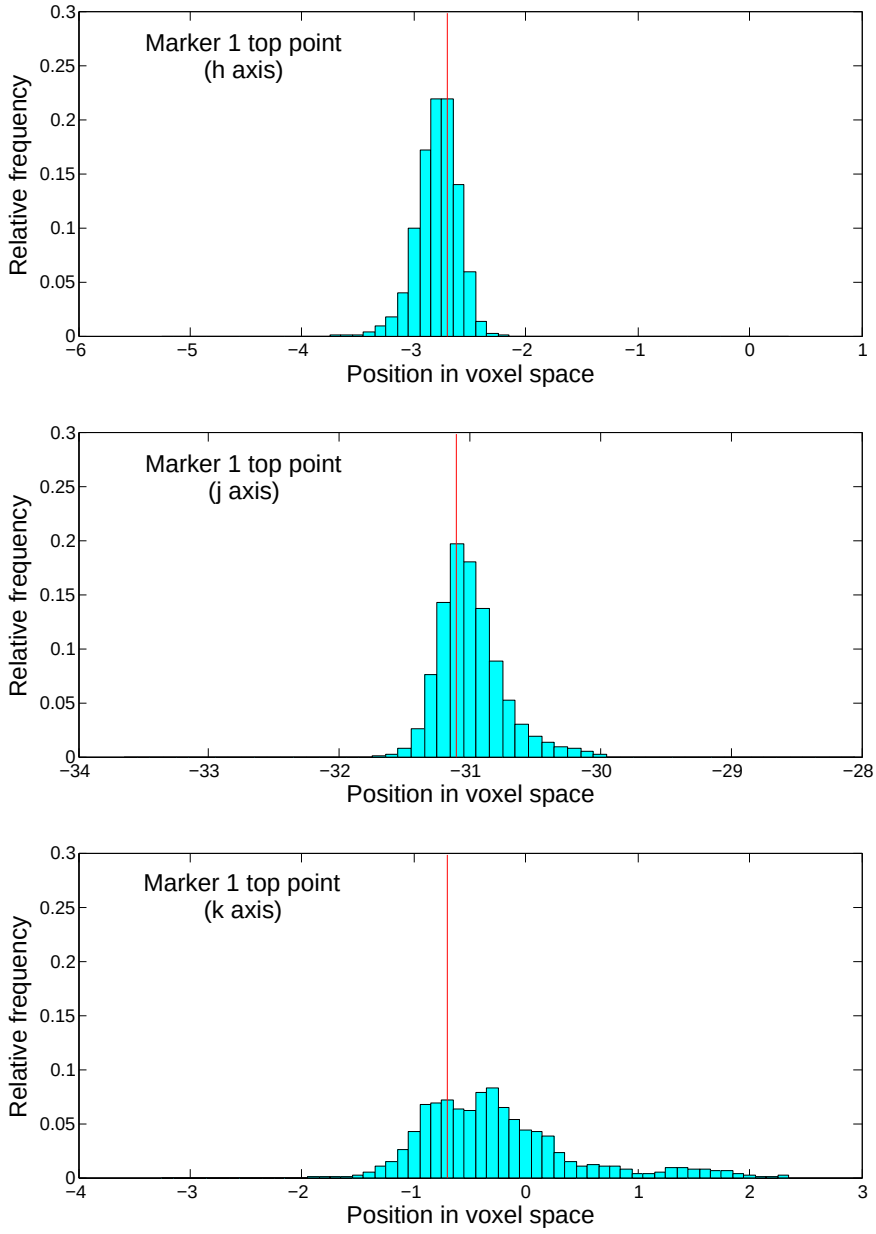


Figure 6.24: Histograms for marker 1 (top point).

Histograms of the samples from the MCMC simulation (after burn-in of 100,000 iterations). The true value is also shown (red).

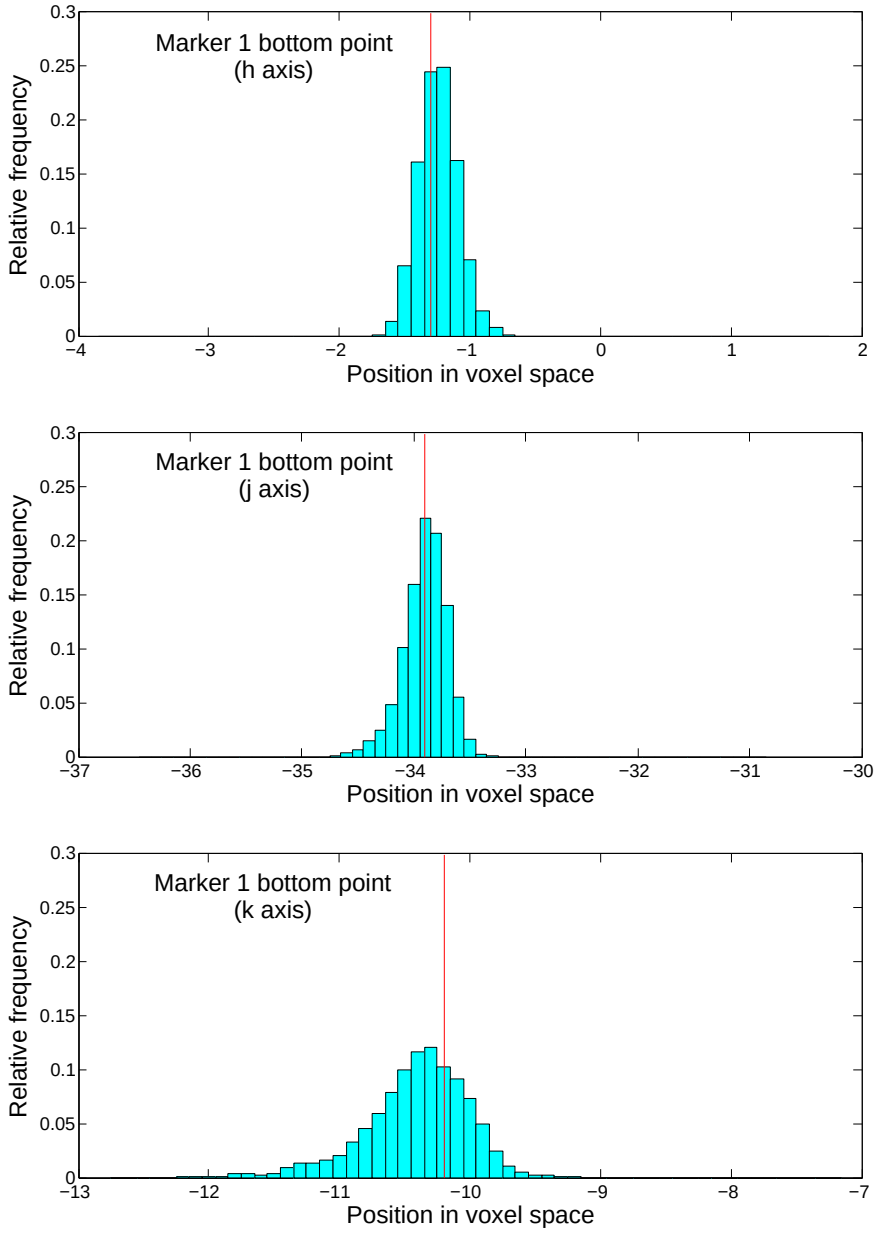


Figure 6.25: Histograms for marker 1 (bottom point).

Histograms of the samples from the MCMC simulation (after burn-in of 100,000 iterations). The true value is also shown (red).

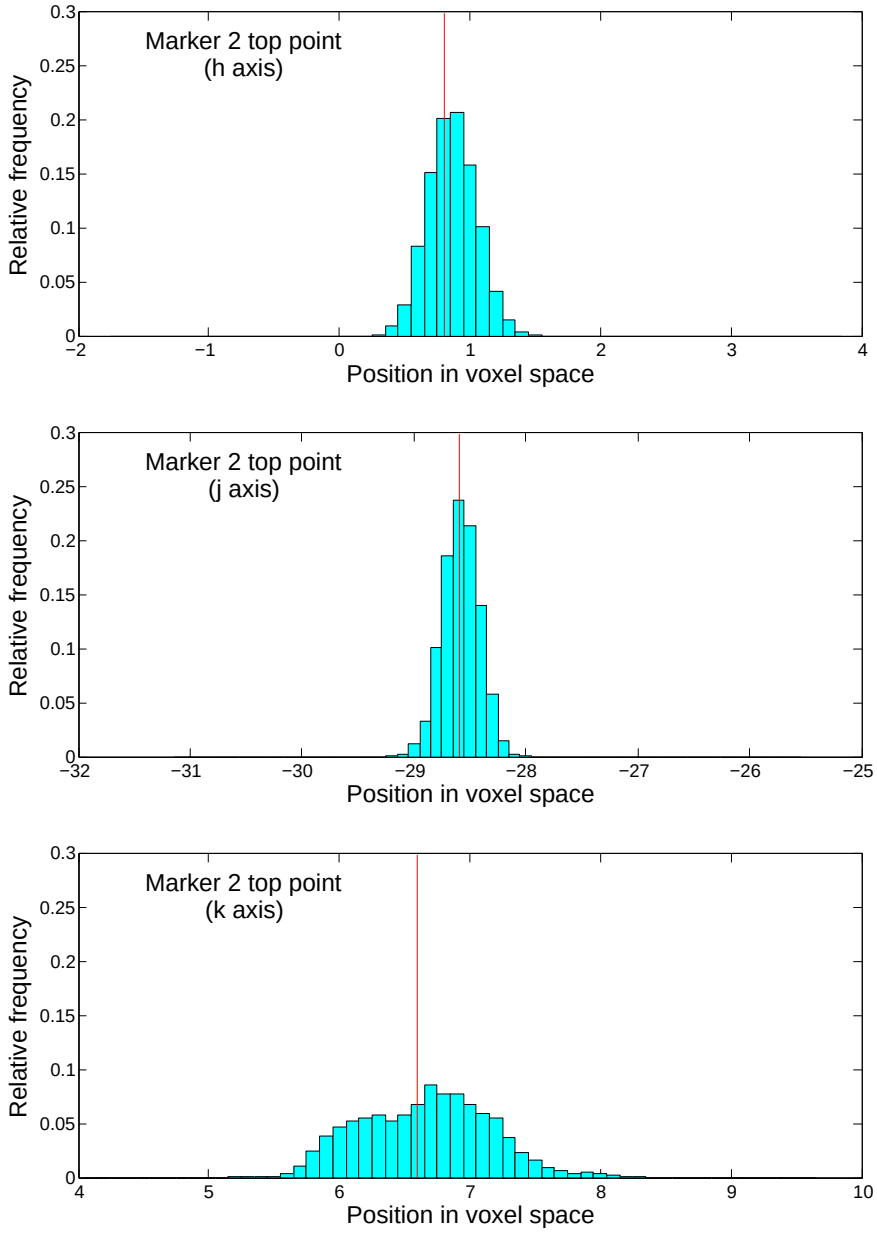


Figure 6.26: Histograms for marker 2 (top point).

Histograms of the samples from the MCMC simulation (after burn-in of 100,000 iterations). The true value is also shown (red).

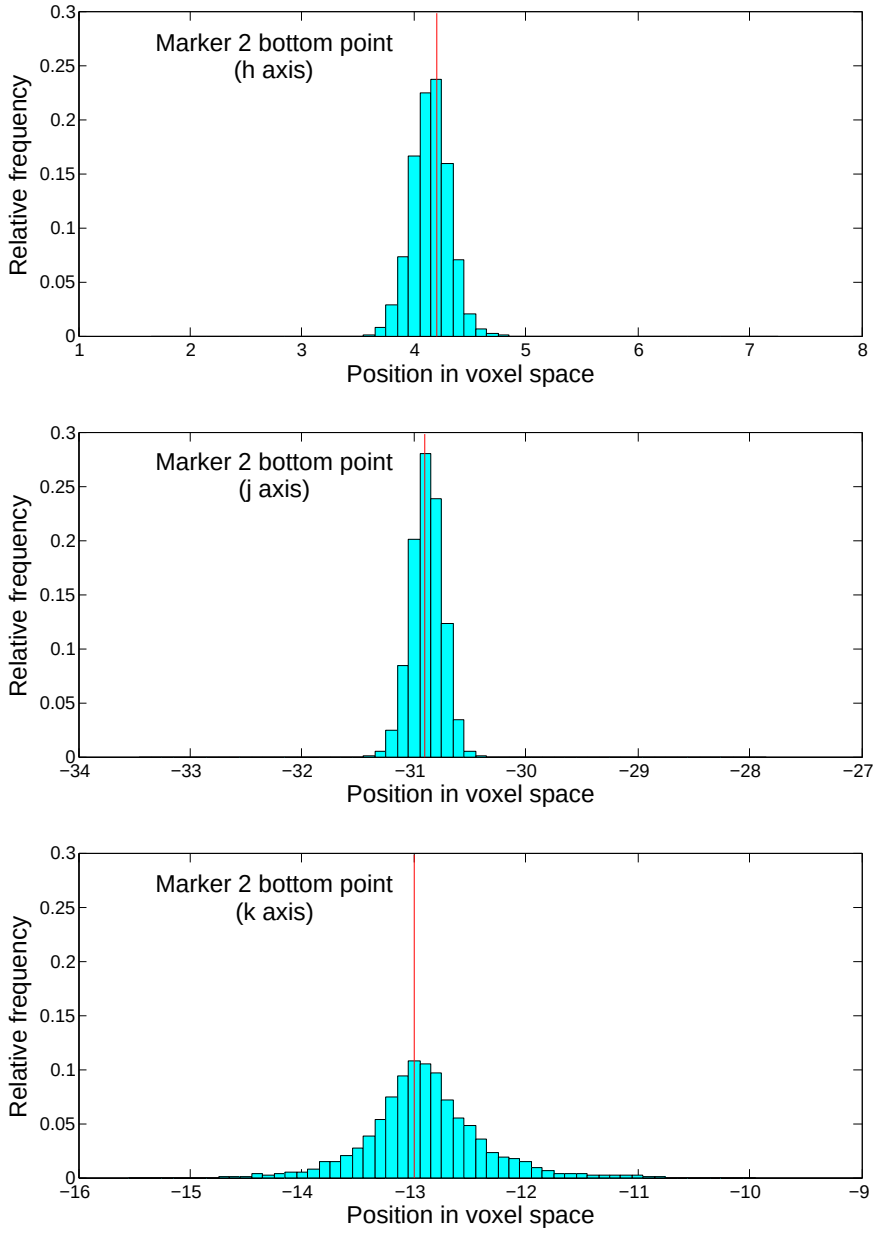


Figure 6.27: Histograms for marker 2 (bottom point).

Histograms of the samples from the MCMC simulation (after burn-in of 100,000 iterations). The true value is also shown (red).

		True value	MPM	95% credible interval
Marker 1 top point	h	-2.7	-2.79	[-3.56 , -2.52]
	j	-31.1	-30.99	[-31.57 , -30.52]
	k	-0.7	-0.26	[-1.91 , 1.32]
Marker 1 bottom point	h	-1.3	-1.24	[-1.65 , -0.99]
	j	-33.9	-33.90	[-34.68 , -33.62]
	k	-10.2	-10.41	[-12.08 , -9.86]
Marker 2 top point	h	0.8	0.87	[0.35 , 1.17]
	j	-28.6	-28.58	[-29.07 , -28.32]
	k	6.6	6.66	[5.39 , 7.43]
Marker 2 bottom point	h	4.2	4.15	[3.68 , 4.41]
	j	-30.9	-30.89	[-31.31 , -30.66]
	k	-13.0	-12.90	[-14.68 , -12.03]

Table 6.1: Results from phantom (with known tissue attenuation values).

the number of samples in $[L - 0.01, L + 0.01]$ and $[U - 0.01, U + 0.01]$ were approximately equal.

These figures and table demonstrate that the posterior mean position of the marker ends is close to the true values. In each case, the credible interval contains the true value of the coordinate. The width of the credible intervals in the $\mathfrak{h}$ and j (i.e. transaxial) directions is approximately 1mm, compared to 2mm or more in the $\mathfrak{k}$ direction. This makes intuitive sense—it is harder to precisely locate the end of the marker in the direction parallel to the marker's central axis.

6.5.4 Modelling both tissue and marker attenuation (phantom data)

Large phantom containing markers

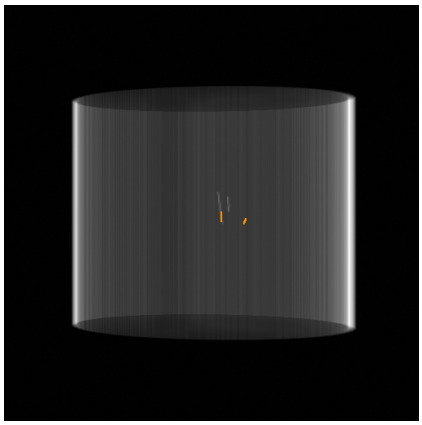
Sections 6.5.2 and 6.5.3 presented the results obtained with, respectively, only the tissue attenuation model (with no markers) and only the marker model (with the tissue attenuation values fixed). In the former, we were able to create a good reconstruction of the tissue attenuation values; in the latter, we accurately located the markers.

In this section, we shall present the results of combining both these processes in the model. In the MCMC sampler, one iteration consisted of updating each of the tissue attenuation values in the voxel grid in a raster fashion, followed by updating one of the marker end coordinates (selected randomly with equal probability).

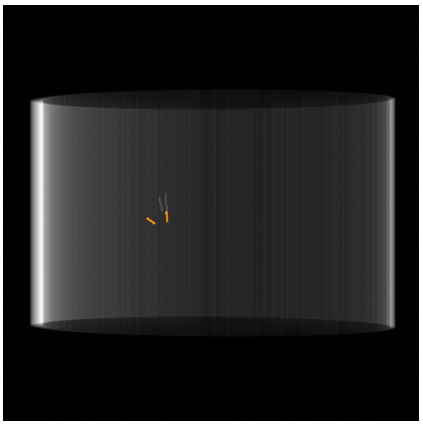
We were not able to successfully reconstruct the structure in the tissue attenuation in the VOI at the same time as locating the markers. Figures 6.28 and 6.29 show an illustration of the output of the sampler, which was run for 500,000 iterations. It is clear that the sampler has not found the markers. Instead, it is sampling a local mode, which can be understood by examining the axial slice (Fig. 6.28c). Where there are really are markers, the tissue attenuation values for the voxels are very high, causing the projections through the voxels to match the measured data. This is similar to the situation where we did not include the marker process. In the voxels where the (modelled) marker has spent a large number of iterations, the MPM tissue attenuation value is very low. This is because the sampled points in the marker are projected to calculate $g_{\text{marker}}(x; u, v, \theta)$; this is added to $g_{\text{tissue}}(x; u, v, \theta)$, which is calculated by ray tracing through the voxels. In order to make the sum of the attenuation due to tissue and due to marker at the point represented by a particular voxel match the measured data (which shows attenuation due to soft tissue only), the tissue attenuation becomes lower—a 'black hole' appears in the tissue attenuation values, filled by the modelled marker.

If we are willing to sacrifice accurate reconstruction of the tissue attenuation values, we can however achieve the (more important) goal of locating the markers. We do this by modifying the prior distribution on the tissue attenuation values so that 'black holes' are less probable. This is done by increasing the parameter β, which controls the penalty for voxel values which do not match their neighbours (Equation 6.1). For high values of β, anything other than a very smoothly varying tissue attenuation has very low probability. We are modifying the prior distribution for the tissue attenuation values so that it assigns very low probability to any configuration with structure in the tissue attenuation values.

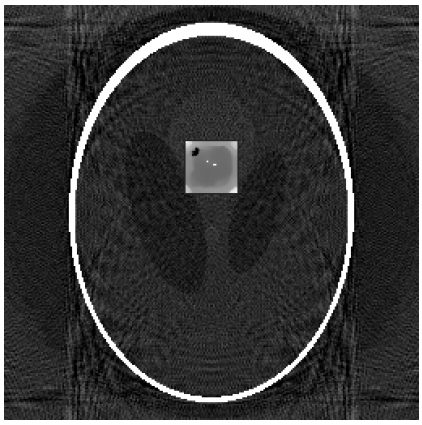
The MPM reconstructions of the phantom in Sections 6.5.2 and 6.5.2 used a value of $\beta = 0.05$, which gives good reconstructions of the tissue attenuation values. The results in Figures 6.28 and 6.29 used $\beta = 0.2$. The effect of this increase



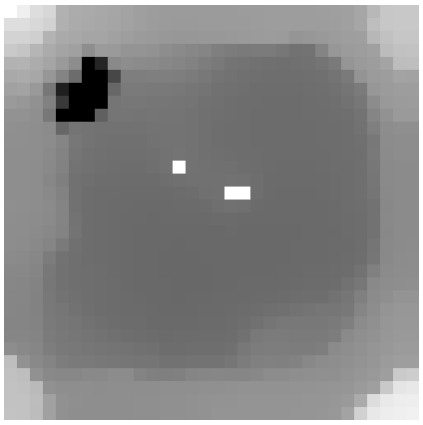
a: Projection of markers sampled from the posterior distribution.



b: Projection of markers sampled from the posterior distribution.



c: Central slice from the MPM tissue attenuation.



d: VOI only.

Figure 6.28: Posterior distribution with both components of model.

Illustration of sampling the posterior distribution when both the tissue attenuation and marker processes are included in the model.

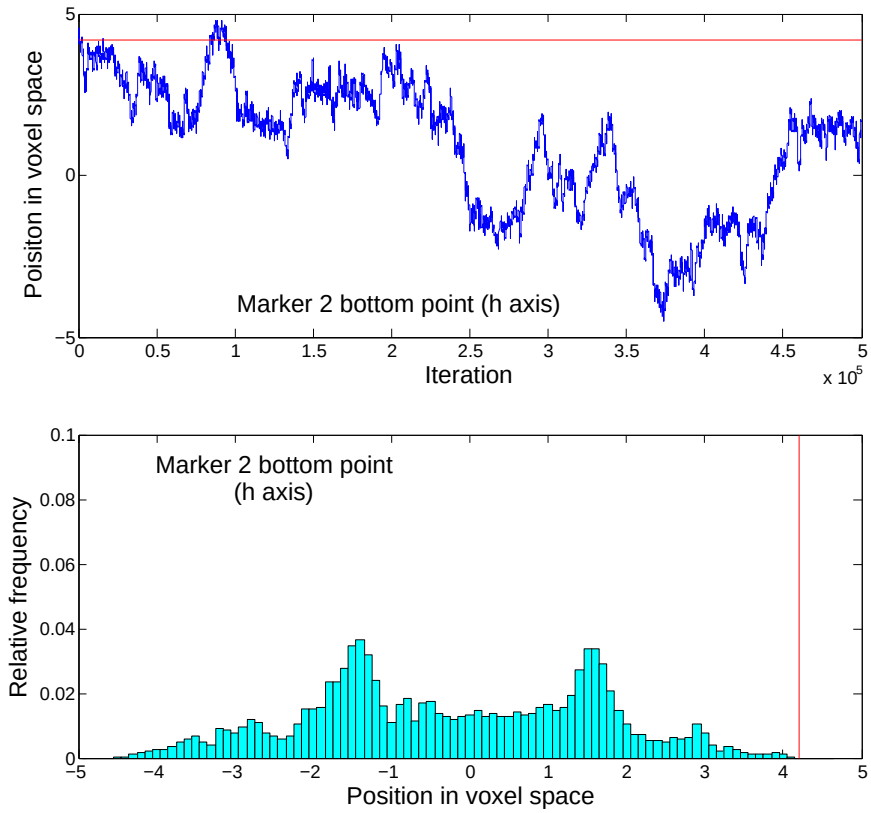


Figure 6.29: Posterior distribution with both components of model.

Illustration of sampling the posterior distribution when both the tissue attenuation and marker processes are included in the model. Shown are a trace showing the sample path for one coordinate and a histogram of the samples for this coordinate.

can be seen in the axial slices—other than the bright markers and dark 'black holes', the voxel value is very smoothly varying.

It was necessary to increase the value of β to approximately 1 in order to avoid the sampler getting stuck in a local mode such as that demonstrated in Figures 6.28 and 6.29. With this high value of β, we do not recover any structure in the tissue attenuation—the MPM reconstructions of the VOI are essentially uniform. Given that we are not recovering structure in the tissue attenuation, we can reduce the number of projections used, with little effect on the ability of the model to locate the markers.

Summaries of the sampler output from four MCMC runs are shown in Tables 6.2 and 6.3. Each experiment (a-d) used 16 images (one in 40), but the four experiments used different data: the first used images $1, 41, 81, \ldots$, the second used images $11, 51, 91, \ldots$ etc. In each case, the MPM coordinate is close to the true value, which is included in the 95% credible intervals. These intervals are wider than those in Table 6.1, especially in the ℓ direction where they are up to 5mm in width.

This demonstrates that in a phantom version of the real problem, where both the tissue attenuation and marker location are unknown, we can reliably locate the markers in the phantom (although we have not been able to recover the structure in the tissue attenuation at the same time). The credible intervals are wider than we might expect, but this may be all the confidence that the data justify.

6.5.5 Modelling the tissue attenuation only (patient data)

We shall now show results of using our model on patient data. We shall use the 'straight marker' data set. The data consist of 626 projection images.

Patient with no markers

In this section, we shall demonstrate the results obtained using the tissue model only (with no markers included in the model) with projection images of part of the patient where there are no markers.

As with the phantom data in section 6.5.2, the posterior distribution of the tissue attenuation values was sampled in a small VOI within the patient. The

		True value	Data set	MPM	95% credible interval
Marker 1	h	-2.7	a	-2.64	[-4.04 , -1.96]
(top point)			b	-2.72	[-3.85 , -2.14]
			c	-2.81	[-4.28 , -1.35]
			d	-2.82	[-4.09 , -2.21]
Marker 1	j	-31.1	a	-31.16	[-33.21 , -30.29]
(top point)			b	-31.13	[-32.25 , -30.47]
			c	-31.02	[-33.11 , -30.21]
			d	-31.03	[-32.33 , -30.32]
Marker 1	k	-0.7	a	-1.15	[-4.26 , 0.77]
(top point)			b	-0.91	[-3.61 , 0.56]
			c	-0.65	[-4.82 , 1.32]
			d	-0.42	[-3.71 , 1.00]
Marker 1	h	-1.3	a	-1.32	[-3.23 , -0.67]
(bottom point)			b	-1.43	[-3.43 , -0.56]
			c	-1.16	[-2.37 , -0.20]
			d	-1.22	[-2.24 , 0.64]
Marker 1	j	-33.9	a	-33.69	[-35.06 , -32.32]
(bottom point)			b	-33.72	[-36.12 , -32.77]
			c	-33.74	[-35.22 , -32.68]
			d	-33.92	[-35.46 , -33.21]
Marker 1	k	-10.2	a	-9.70	[-12.39 , -6.59]
(bottom point)			b	-9.32	[-13.42 , -6.83]
			c	-9.85	[-12.00 , -8.54]
			d	-10.25	[-13.28 , -8.17]

Table 6.2:
Results from phantom with unknown tissue attenuation values (marker 1).
The table shows the results from four replications of the experiment (a-d).

		True value	Data set	MPM	95% credible interval
Marker 2	h	0.8	a	0.94	[-0.33 , 1.52]
(top point)			b	0.92	[-0.38 , 1.68]
			c	0.91	[-0.20 , 1.43]
			d	0.92	[-0.34 , 1.48]
Marker 2	j	-28.6	a	-28.68	[-29.56 , -28.19]
(top point)			b	-28.62	[-29.71 , -28.10]
			c	-28.75	[-29.85 , -28.25]
			d	-28.74	[-29.59 , -28.17]
Marker 2	k	6.6	a	6.10	[1.99 , 8.29]
(top point)			b	6.14	[0.26 , 8.55]
			c	5.87	[3.07 , 7.35]
			d	5.99	[3.15 , 8.15]
Marker 2	h	4.2	a	4.10	[3.12 , 4.61]
(bottom point)			b	4.08	[3.24 , 4.56]
			c	4.08	[2.78 , 4.71]
			d	4.15	[3.19 , 4.78]
Marker 2	j	-30.9	a	-30.85	[-31.62 , -30.43]
(bottom point)			b	-30.81	[-31.73 , -30.37]
			c	-30.81	[-31.84 , -30.38]
			d	-30.82	[-31.57 , -30.44]
Marker 2	k	-13.0	a	-12.83	[-15.01 , -9.95]
(bottom point)			b	-12.57	[-15.02 , -9.63]
			c	-12.57	[-15.01 , -10.06]
			d	-12.94	[-15.02 , -10.88]

Table 6.3:
Results from phantom with unknown tissue attenuation values (marker 2).
The table shows the results from four replications of the experiment (a-d).

whole reconstruction $\mathcal{R}$ was 416mm×416mm×144mm. We consider the tissue attenuation on a voxel grid, with voxel pitch 1mm. The VOI was $55 \times 50 \times 20$ voxels in size.

The voxel values in the region outside the VOI were taken to be the values obtained from an FDK reconstruction using all 626 projection images. For the sampler, one in ten of the projection images (i.e. 62 images) were used. The sampler ran for 50,000 iterations, with the first 20,000 discarded as burn-in. The value of β in the prior for the tissue attenuation was 0.05.

As for the phantom data, we need to rescale the FDK reconstruction so that the projected voxel values can be compared with the measured data. An ROI slightly below the markers was chosen, as shown in Fig. 6.30. This figure also demonstrates the severity of the artefacts in the FDK reconstructions using the real data.

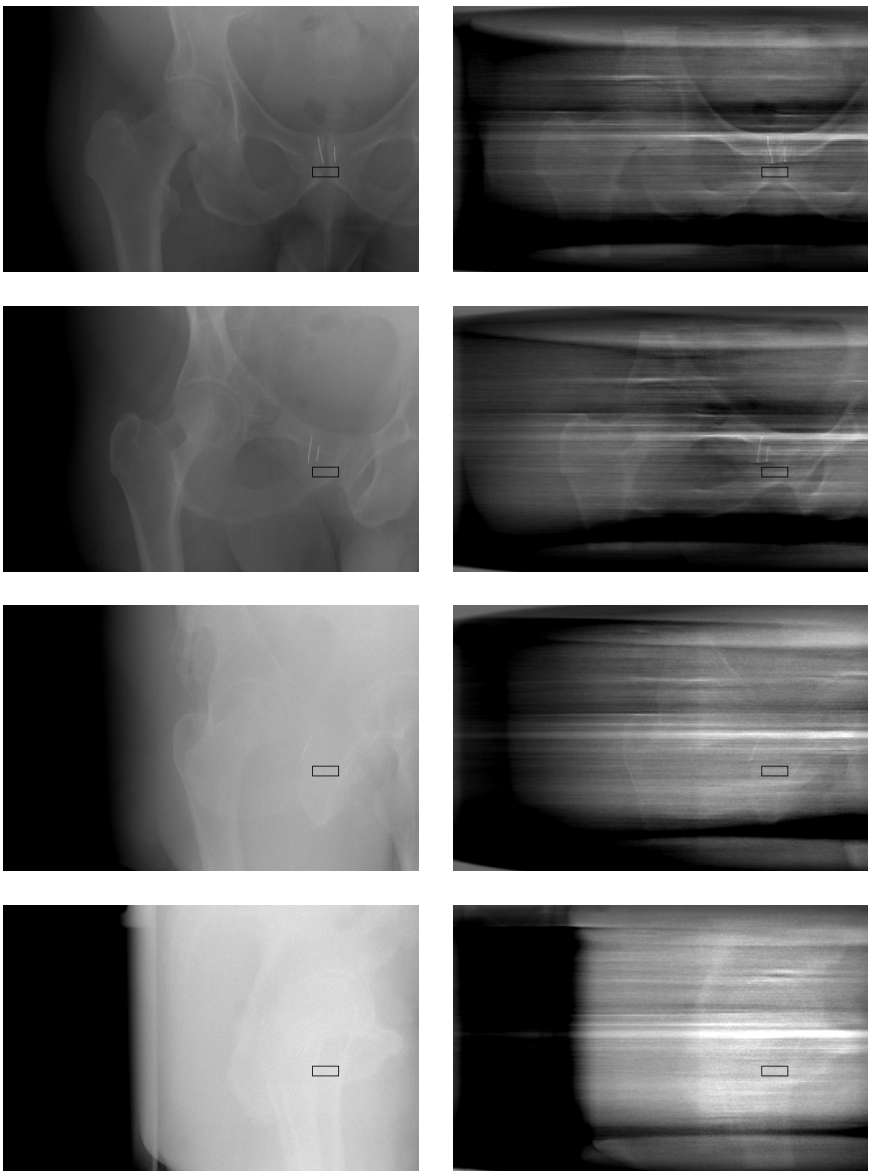
The VOI contains bone and soft tissue. Unlike with the phantom data, we do not have a ground truth with which to compare. Fig. 6.31 shows a comparison between axial slices of the FDK reconstruction, and of the MPM image. The bony structure in the VOI can be seen both reconstructions. The structure is clearer in the FDK reconstruction, although it is not known which is closer to the truth.

Modelling markers as tissue

We now consider the posterior distribution of the tissue attenuation values (with no modelled markers) in a part of the patient which does contain markers, as in Section 6.5.2. Fig. 6.32 shows axial slices of the FDK reconstruction and the MPM image of a VOI containing the markers. For this analysis, the VOI was $25 \times 30 \times 26$ voxels in size. The markers can be clearly identified in both reconstructions, and the streak artefact between the two markers has been reduced in the MPM reconstruction.

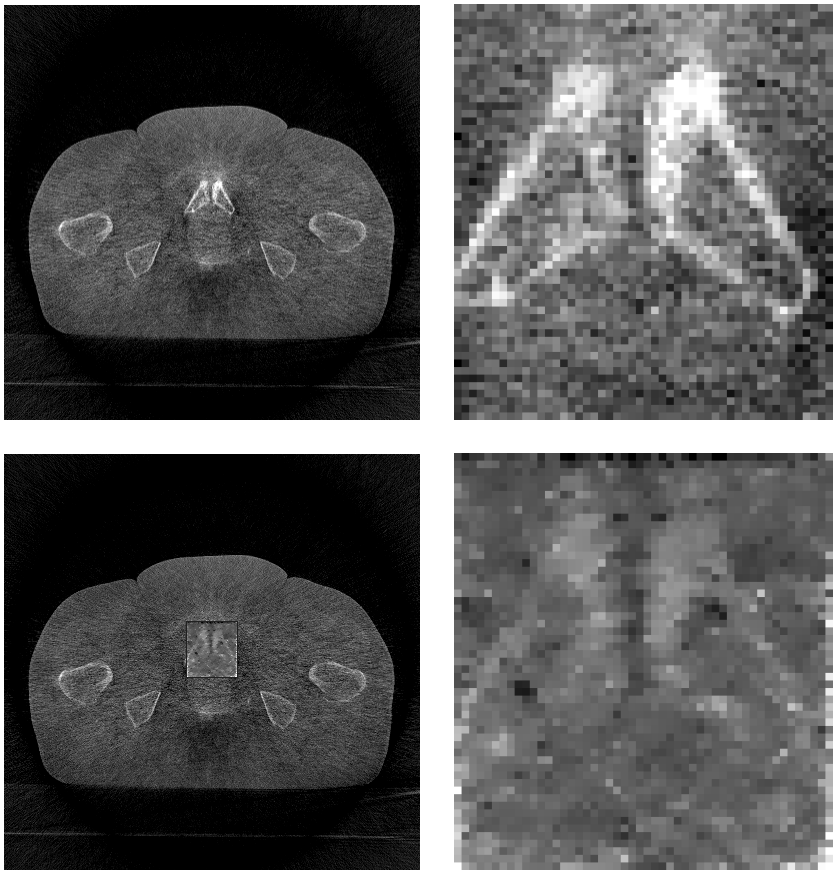
6.5.6 Modelling both tissue and marker attenuation (patient data)

Table 6.4 shows the MPM values and estimates of 95% credible intervals obtained by running the sampler with both the tissue attenuation and marker processes-included in the model. We used a high value of the tissue prior parameter:



Original data. Projection through the reconstruction.

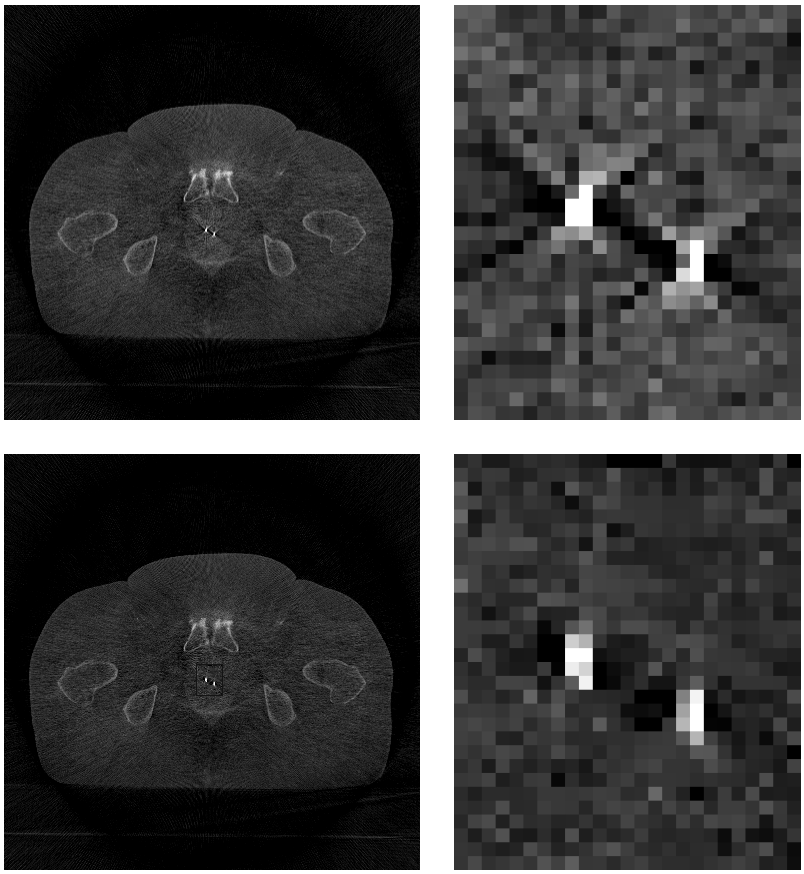
Figure 6.30: The ROI used for rescaling the reconstruction. The ROI is shown on four of the images.



Central slice. VOI only.

Figure 6.31: FDK and MPM reconstructions with no markers.

The images show slices from a reconstruction of part of the patient not containing markers. Top: The FDK reconstruction. Bottom: MPM estimate of tissue attenuation values in a VOI embedded in the FDK reconstruction.



Central slice. VOI only.

Figure 6.32: FDK and MPM reconstruction of patient with markers.

The images show slices from a reconstruction of part of the patient containing markers. Top: The FDK reconstruction. Bottom: MPM estimate of tissue attenuation values in a VOI embedded in the FDK reconstruction.

$\beta = 0.5$. Fig. 6.33 shows central slices of the MPM tissue values. This is very smooth compared to those in Fig. 6.32 because of the higher β value used. Fig. 6.34 shows the 95% credible intervals (in three dimensions) projected and overlaid on the original data for four projection images.

These figures and table demonstrate that the estimates of the marker location are accurate for one of the markers (marker 1 in Table 6.4; the longer marker in Fig. 6.34). However, the estimates for the other (shorter) marker are not accurate—they describe a marker which is much longer than the true marker. This behaviour was not seen in all realisations (different seeds and/or subsets of projections)—most resulted in accurate estimates—but it was not unusual.

We postulate that there is a local mode representing a longer marker, caused by the artefacts in the reconstruction used to estimate the values of X_{tissue} in the volume outside the VOI. When we calculate $g_{\text{tissue}}(x; u, v, \theta)$ (by ray-tracing through the whole volume), the artefacts cause the calculated values to be higher or lower than the truth. This can be seen in Fig. 6.30, where the artefacts in the FDK reconstruction cause the stripy appearance of the projected values on the right-hand side. In calculating the likelihood, we compare the value of $g_{\text{tissue}}(x; u, v, \theta)$ to the measured pixel value. If artefacts have caused $g_{\text{tissue}}(x; u, v, \theta)$ to be lower than it should be, a better fit to the projection data may be obtained by a realisation of the marker where it appears on the path to this pixel. In this case, we compare the sum of $g_{\text{marker}}(x; u, v, \theta)$ and $g_{\text{tissue}}(x; u, v, \theta)$ to the measured pixel value. Once the marker is longer than 15mm, the prior encourages it to reach 20mm.

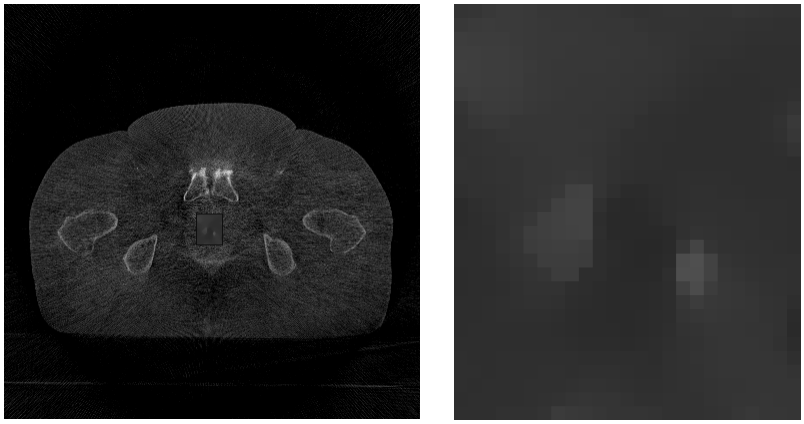
6.5.7 Modelling the marker attenuation only (patient data)

The artefacts in the initial FDK reconstruction have hampered our ability to jointly estimate the tissue attenuation values and the marker locations. In this section, we shall revert to our primary goal of estimating the marker locations only.

In Section 6.5.3, we demonstrated for the large phantom that we could accurately find the marker locations if we had a reconstruction of the phantom with no markers. This marker-free reconstruction is easy to obtain in the case of the phantom. In Chapter 5, we showed how we could obtain a reconstruction in which the markers had been largely removed and the artefacts reduced for the patient

		MPM	95% credible interval
Marker 1 top point	h	7.80	[7.26 , 8.09]
	j	-9.66	[-11.04 , -8.96]
	k	-3.04	[-4.75 , -2.14]
Marker 1 top point	h	4.96	[4.31 , 5.30]
	j	-21.38	[-23.14 , -20.59]
	k	14.04	[12.30 , 14.99]
Marker 2 bottom point	h	-1.72	[-3.65 , -0.86]
	j	-17.52	[-19.84 , -15.95]
	k	-0.67	[-3.64 , 1.61]
Marker 2 bottom point	h	-2.76	[-4.24 , -2.09]
	j	-24.86	[-28.10 , -23.58]
	k	14.67	[12.15 , 16.54]

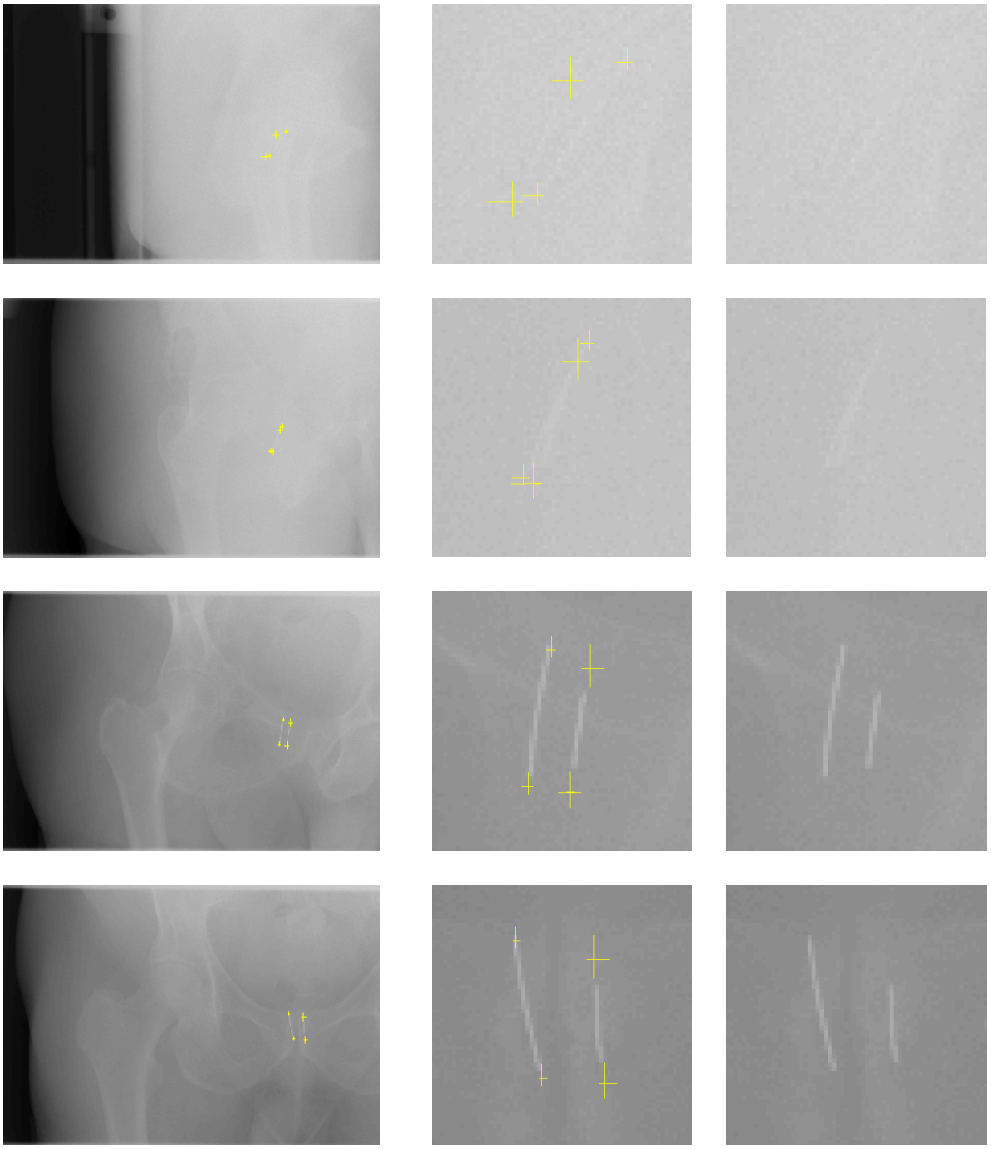
Table 6.4: Results from patient (with unknown tissue attenuation values).



Central slice. VOI only.

Figure 6.33: Reconstruction using both parts of model.

The images show central axial slices from the reconstruction (the same slice as Fig. 6.32), showing part of the patient with markers in. The values shown are the tissue attenuation values.



Whole projection. Close-up of the markers. Original data only.

Figure 6.34: Credible intervals using both parts of model.

The 95% credible intervals for each of the coordinates of each of the end points (yellow), projected and overlaid on four of the original images.

data: we processed the projection images to remove the markers, and used the processed images to create an FDK reconstruction.

This suggests a way of using the 3D model described in this chapter to estimate the location of the markers without simultaneously estimating the tissue attenuation values: use the marker- and artefact-free reconstruction obtained by performing the analysis in Chapter 5, in the same way that we used the reconstruction of the phantom without markers in Section 6.5.3. We did this, but found that this also suffered from the same problem as before—according to the posterior distribution, the most probable locations for the markers were in dark areas caused by artefacts in the FDK reconstruction (with the artefacts coming from the femoral heads, rather than the markers).

Instead we have used another approach seen in Chapter 5: instead of using a projection through the reconstruction as the 'background' in which to fit the marker in each projection, we use a constant background value. In Chapter 5, our model included a background pixel value in each of the 2D ROIs. In the 3D model in this chapter, the value of $g_{\text{tissue}}(x; u, v, \theta)$ serves the same purpose. In each case, if the modelled marker is projected into a particular pixel, we add a certain amount (depending on the model) to the background value, and then use this as the mean value in our calculation of the likelihood of the data for that pixel. We use the FDK reconstruction to create an estimate of X_{tissue} in the patient outside the VOI, which we use to calculate $g_{\text{tissue}}(x; u, v, \theta)$. The artefacts in the FDK reconstruction mean that this does not reflect the truth.

The approach we use is to determine an ROI in each of the 2D projection images, such that every point in the VOI is projected into this ROI. We calculate an estimate $g_{\text{tissue}}(x; u, v, \theta)$ for all pixels in this ROI by calculating the median of the pixel values in the data at the edge of this ROI (as we did for the the ROIs in Chapter 5). We no longer need to scale the FDK reconstruction to match the measured data—using this data-dependent value as an estimate for $g_{\text{tissue}}(x; u, v, \theta)$ achieves the same end.

We then proceed as before, by sampling from the posterior distribution of the marker parameters given the data, $\pi(X_{\text{tissue}}|y)$. The likelihood of the data is normal, with the mean being the sum of the the estimated background value $g_{\text{tissue}}(x; u, v, \theta)$ and the contribution from the marker $g_{\text{marker}}(x; u, v, \theta;)$ (which is,

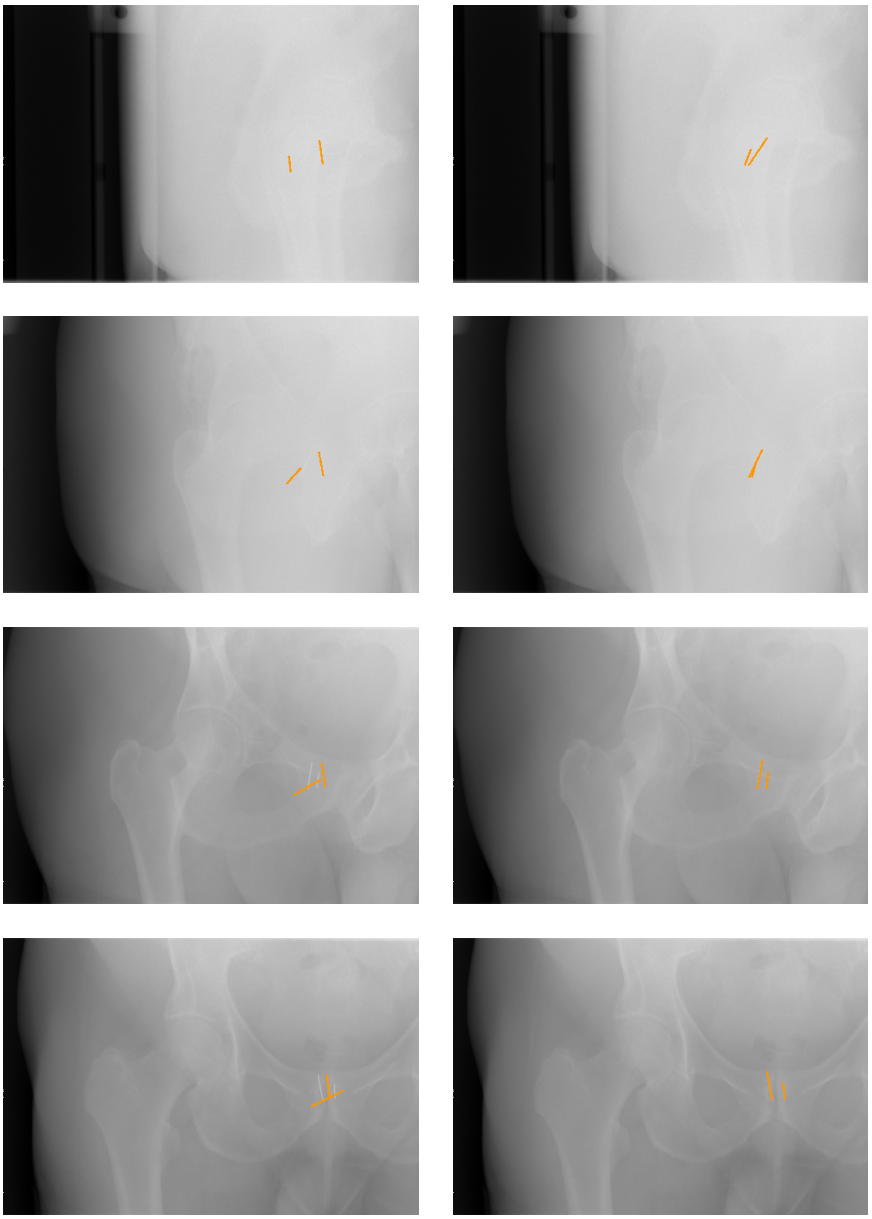
		MPM	95% credible interval
Marker 1 top point	h	-1.93	[-2.18 , -1.82]
	j	-18.62	[-19.47 , -18.58]
	k	1.27	[-0.04 , 2.48]
Marker 1 bottom point	h	-2.45	[-2.68 , -2.31]
	j	-23.25	[-23.68 , -22.94]
	k	12.09	[11.33 , 12.41]
Marker 2 top point	h	8.15	[7.70 , 8.25]
	j	-8.50	[-10.07 , -8.29]
	k	-5.00	[-6.75 , -3.58]
Marker 2 bottom point	h	5.15	[4.94 , 5.32]
	j	-20.71	[-21.25 , -20.31]
	k	13.13	[11.17 , 13.47]

Table 6.5:
Results from patient (with no estimation of tissue attenuation values).

as before, proportional to the volume of the marker within the pyramid of 3D space which is projected to a particular pixel).

Figures 6.35 and 6.36 show representative results from one MCMC run. In Fig. 6.35, projections of the initial position are shown, along with projections of a sample from the posterior distribution. The true marker positions have been found from a fairly arbitrary start point (although one in which the projection of the initial positions intersects the projection of the true position in some projections). Projections of 95% credible intervals (which include the true end) are shown in Fig. 6.36. Traces of the MCMC sampler are shown in Figures 6.37 to 6.40; histograms of the samples are shown in Figures 6.41 to 6.44, and Table 6.5 gives the numerical results for this experiment.

This analysis has succeeded in locating the markers. The widths of the credible intervals in the ℓ (cranio-caudal) direction are again larger than in other directions. However, unlike in the phantom images, the intervals in the j direction are also fairly wide. This is the anterio-posterior direction. Information about the location of the markers in this direction is derived from lateral images, where the contrast

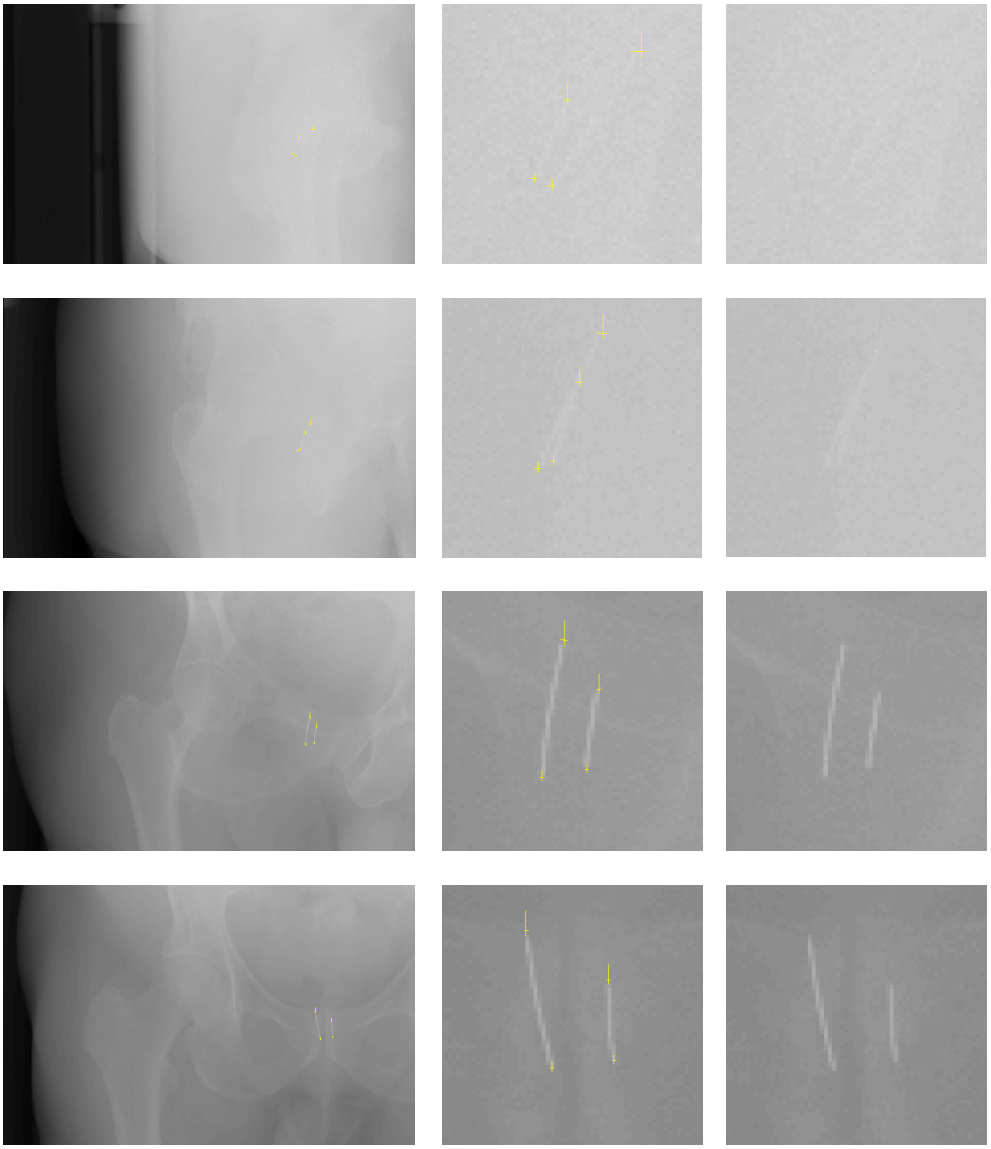


Projections of the intial marker.

Projections of a marker sampled from the posterior distribution.

Figure 6.35: Projections of initial and sampled markers.

The markers have been sampled from a model where the total tissue attenuation has been estimated by the median of the pixel values at the edge of the ROI.



Whole projection. Close-up of the markers. Original data only.

Figure 6.36: Credible intervals using tissue attenuation only.

The 95% credible intervals for each of the coordinates of each of the end points (yellow), projected and overlaid on four of the original images.

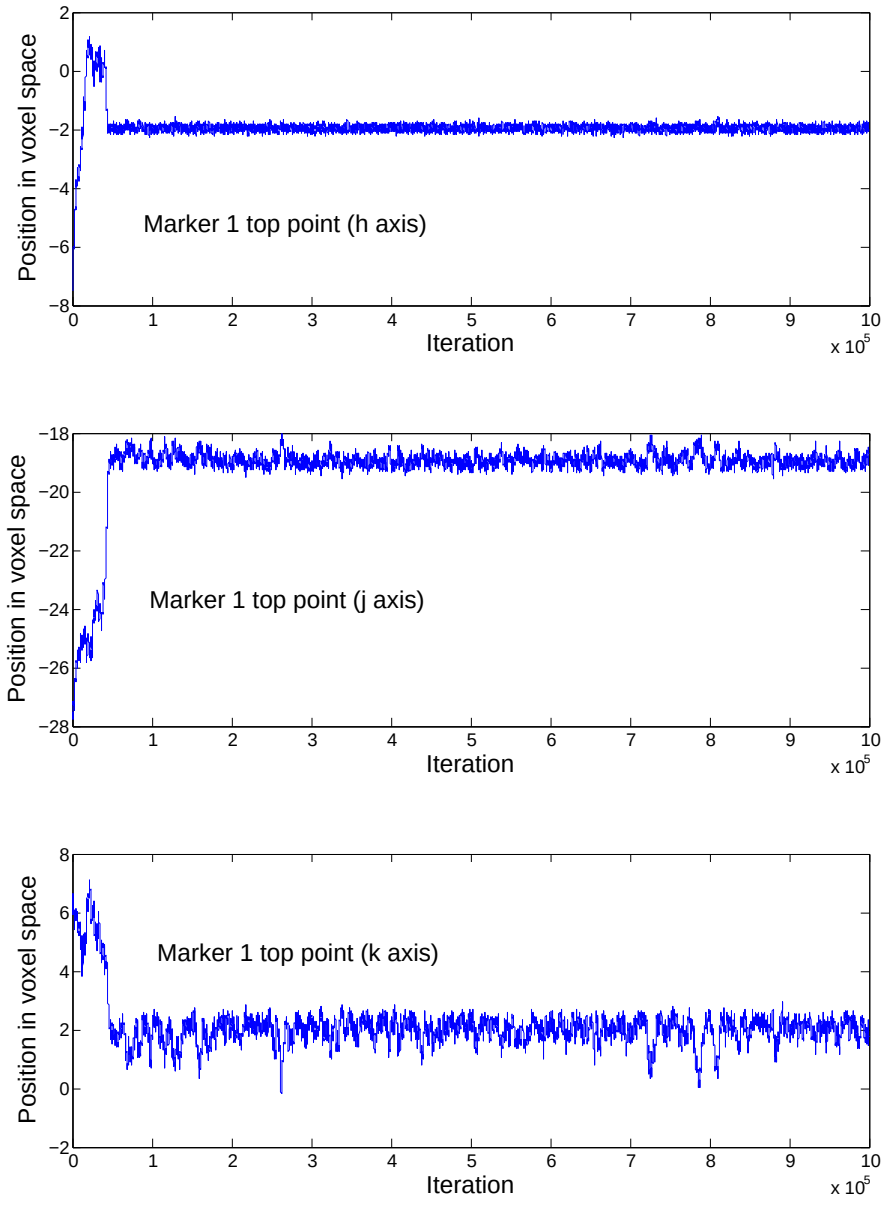


Figure 6.37: The sample path for marker 1 (top point).

The sample paths of each of the three coordinates in the MCMC simulation.

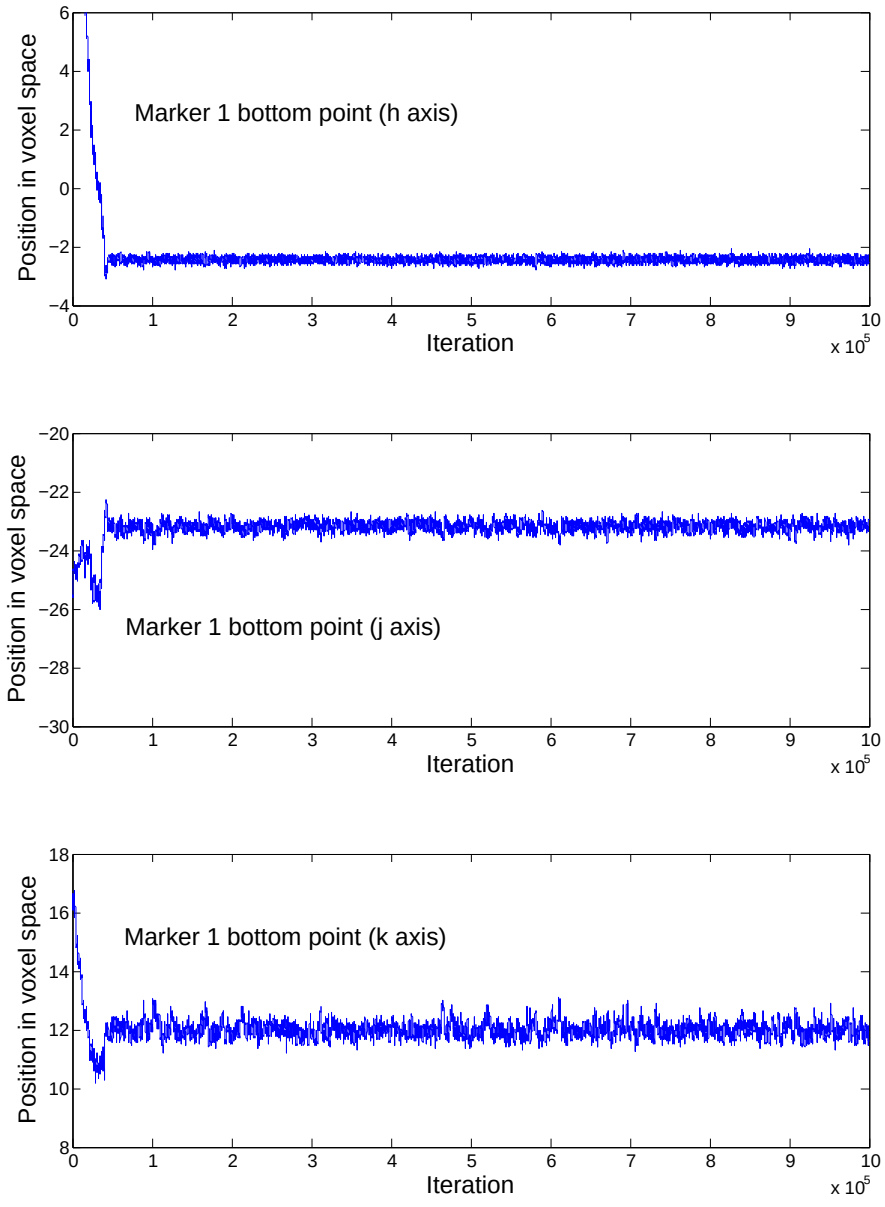


Figure 6.38: The sample path for marker 1 (bottom point).

The sample paths of each of the three coordinates in the MCMC simulation.

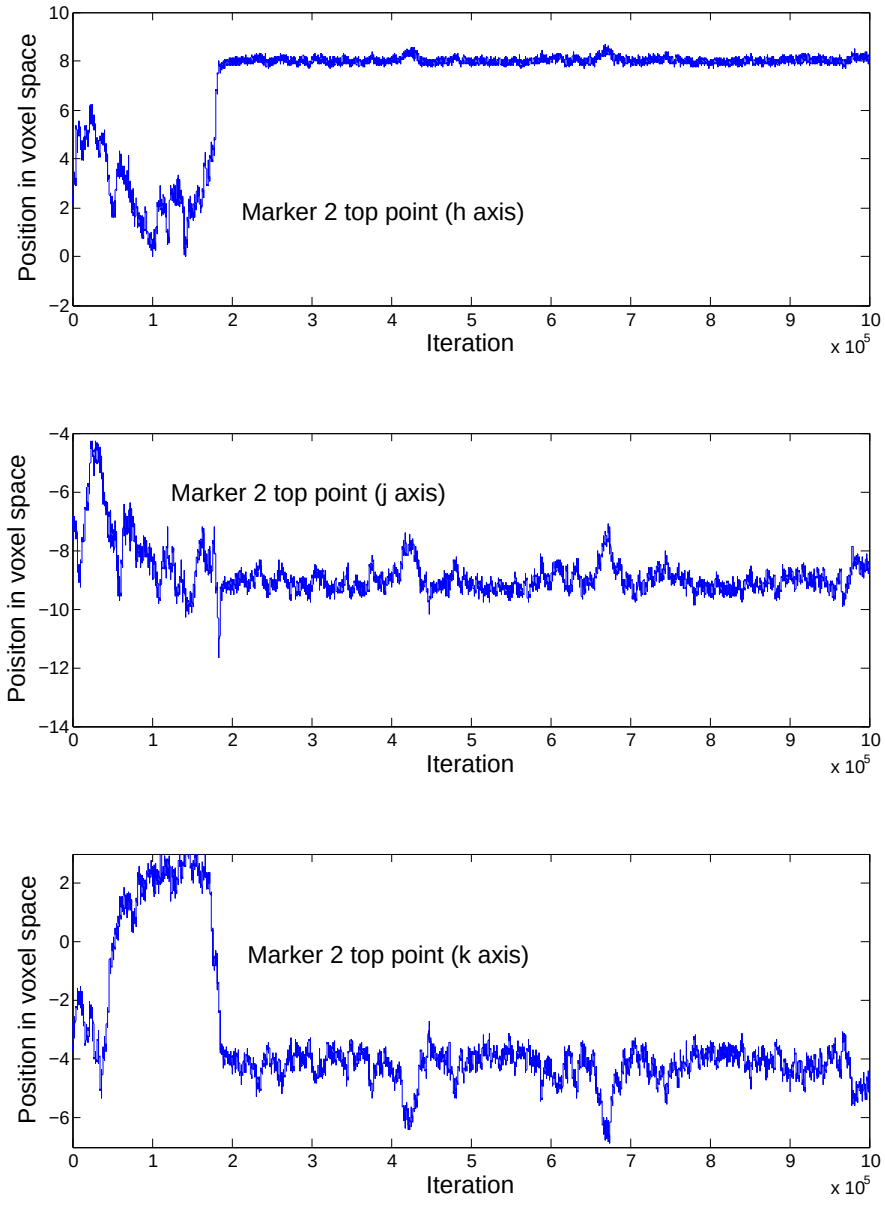


Figure 6.39: The sample path for marker 2 (top point).

The sample paths of each of the three coordinates in the MCMC simulation.

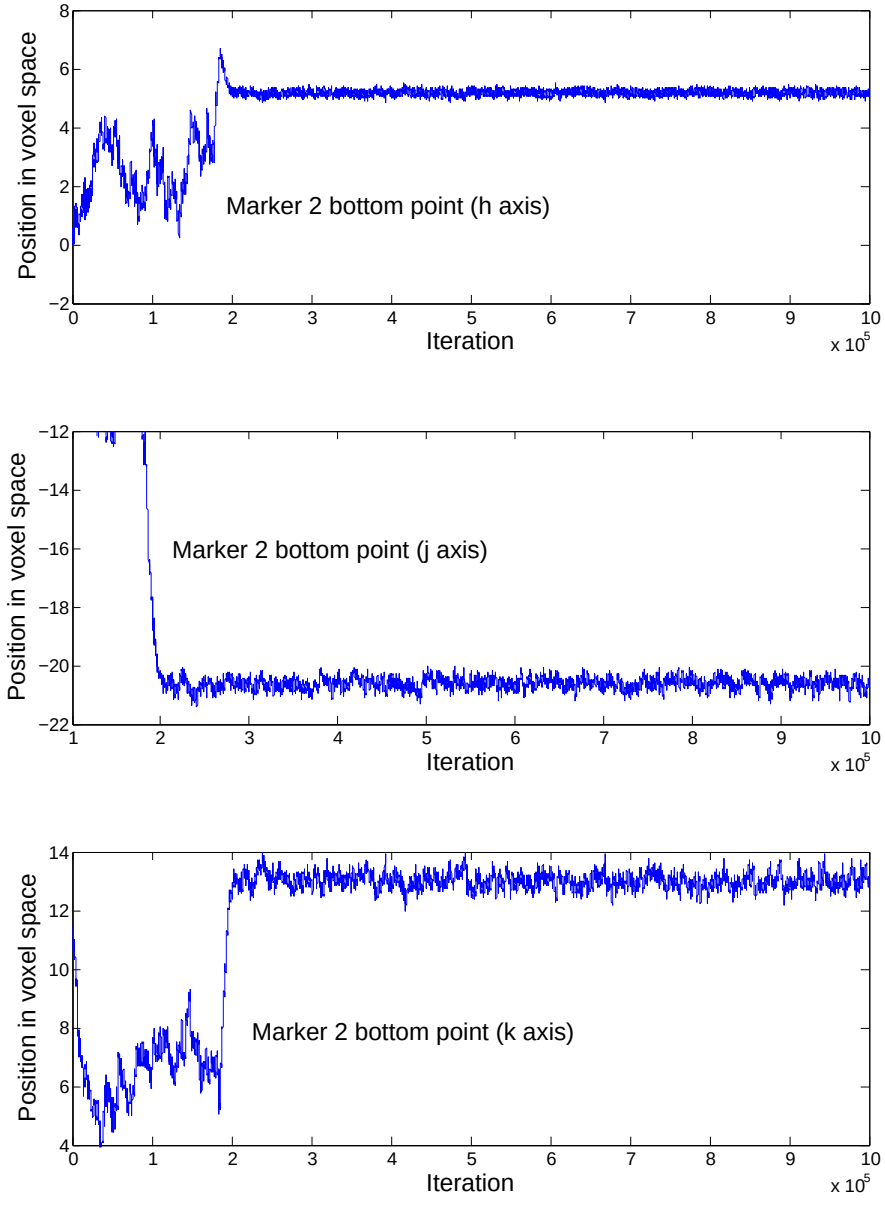


Figure 6.40: The sample path for marker 2 (bottom point).

The sample paths of each of the three coordinates in the MCMC simulation.

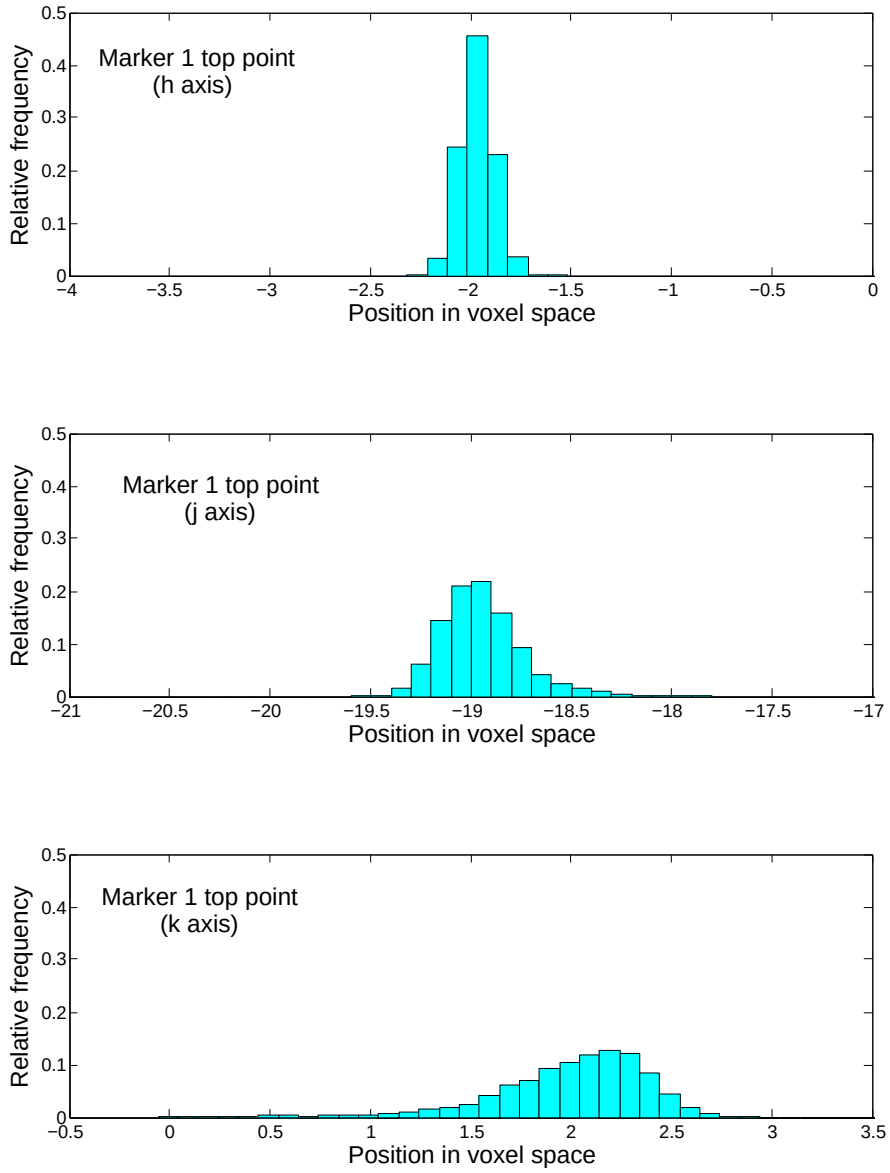


Figure 6.41: Histograms for marker 1 (top point).

Histograms of the samples from the MCMC simulation (after burn-in of 200,000 iterations).

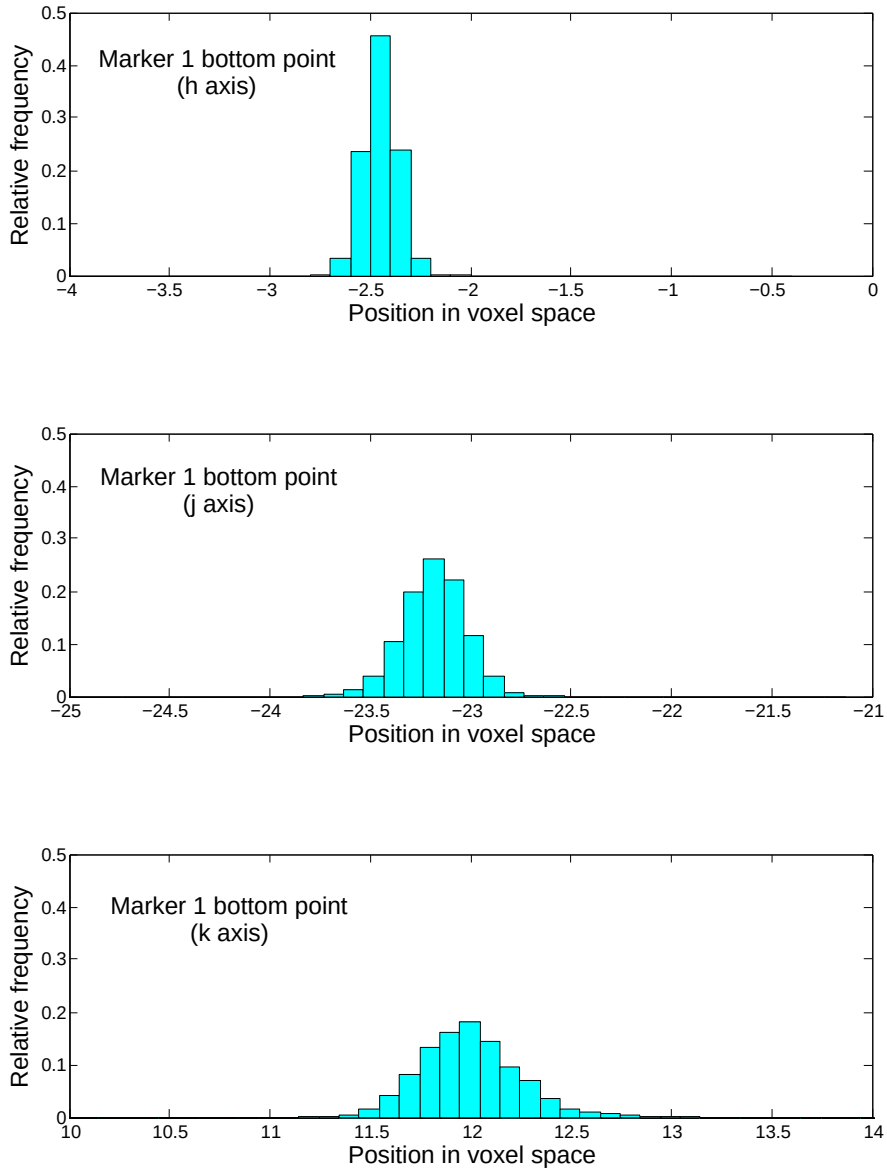


Figure 6.42: Histograms for marker 1 (bottom point).

Histograms of the samples from the MCMC simulation (after burn-in of 200,000 iterations).

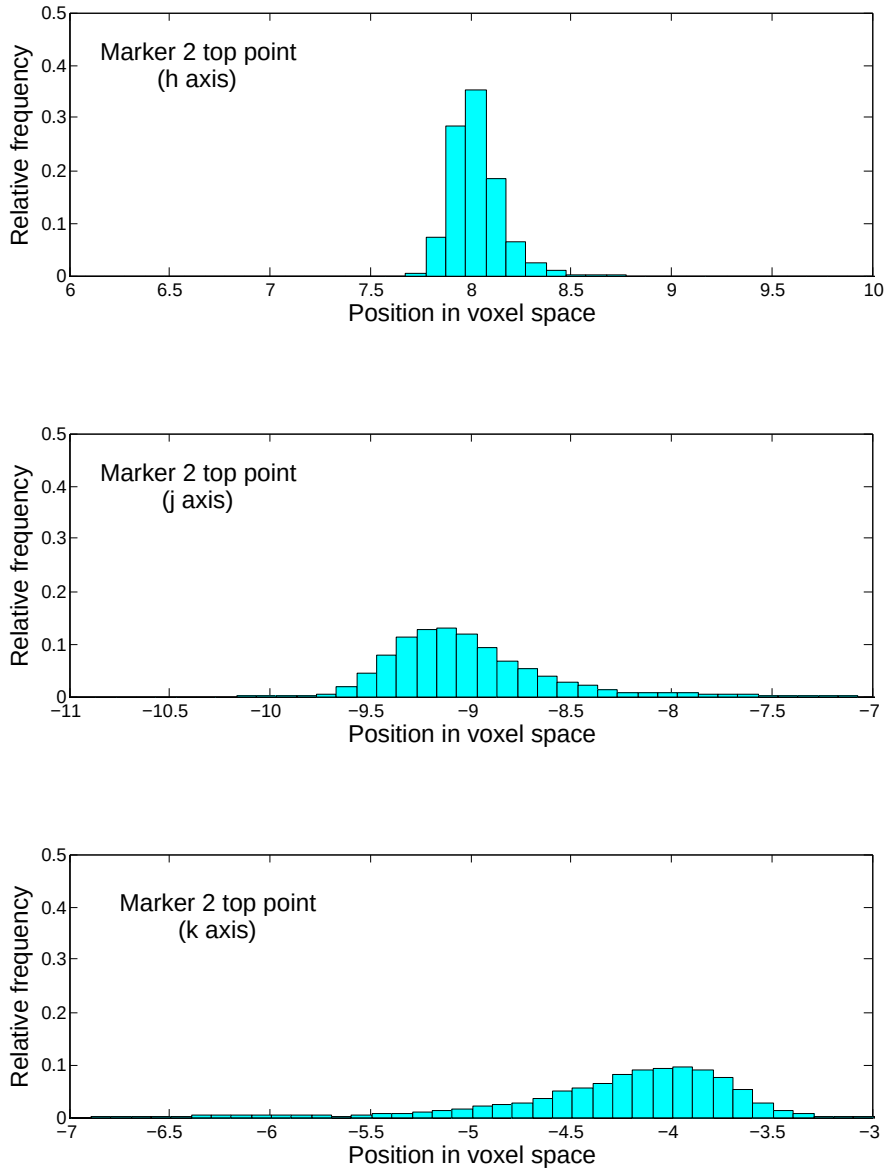


Figure 6.43: Histograms for marker 2 (top point).

Histograms of the samples from the MCMC simulation (after burn-in of 200,000 iterations).

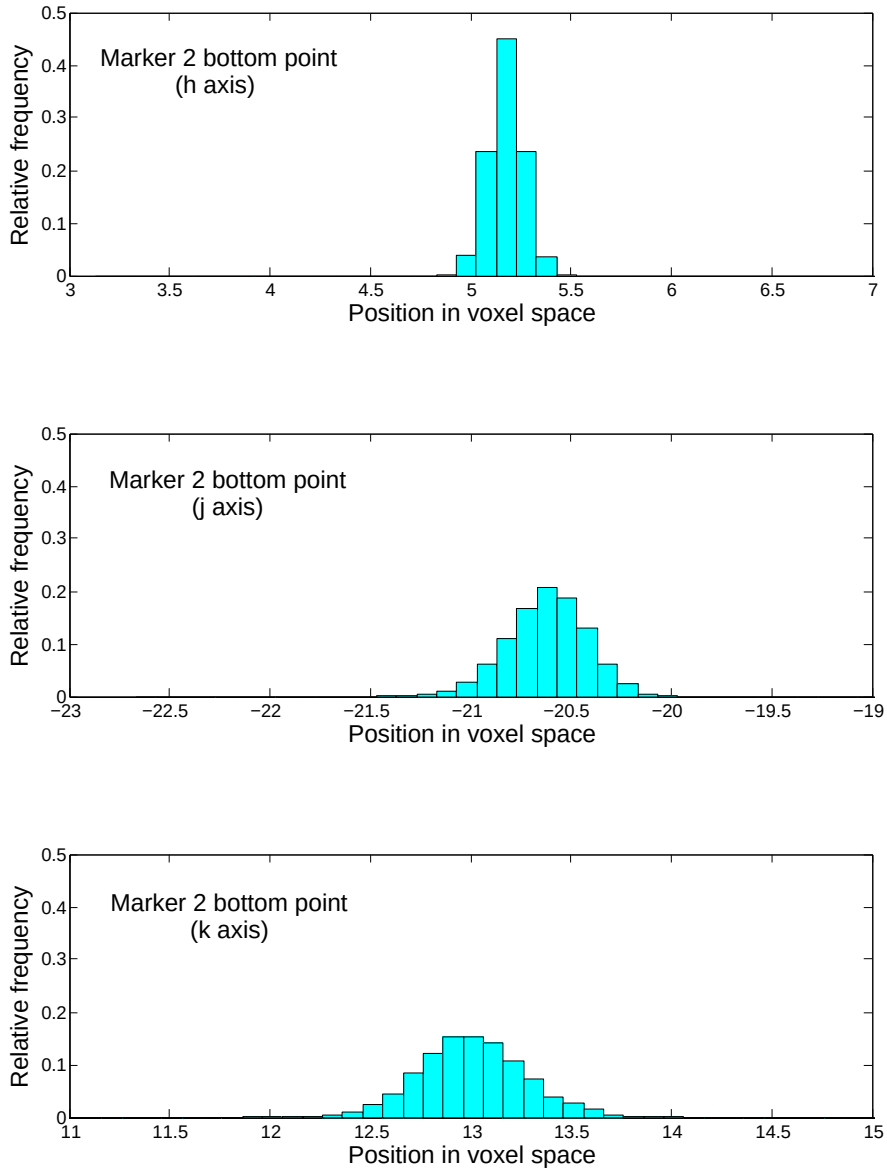


Figure 6.44: Histograms for marker 2 (bottom point).

Histograms of the samples from the MCMC simulation (after burn-in of 200,000 iterations).

is very low due to the large amount of scattered radiation. This was not included in our forward model when generating the projection images from the phantoms.

The MPM top point of each marker is slightly above the true position. The reason for this has not been determined. The distances between the MPM end points are 12mm and 22mm, indicating that increasing the concentration of the prior probability at the known true lengths of 10mm and 20mm might be useful.

By using the median value from the measured data as our estimate of the tissue attenuation along the path to the pixel, we have successfully located the ends of the markers in 3D. This has been at the expense of estimating the tissue attenuation values.

6.6 Discussion

In this chapter, we have described a 3D model for the markers. They are parameterised as cylinders in 3D space, in a way which makes extending the model to more complicated shapes very natural (see Section 6.7). We have also defined a model for the tissue attenuation, with the two processes being assumed independent. We were able to include either the tissue attenuation X_{tissue}, the marker parameters X_{marker}, or both in the model.

We applied the model using the tissue attenuation X_{tissue} only to phantom data. We demonstrated that the MPM reconstructions of the tissue attenuation values were close to the truth, for phantoms with or without markers.

We also examined the performance of the model for the marker parameters X_{marker} alone. Using phantom data enabled us to create matching phantoms with and without markers. We were able to use the reconstruction created from the phantom without markers to give us tissue attenuation values to use in an experiment using the phantom with markers. We considered the tissue attenuation values as fixed, and demonstrated that we could use the marker model to accurately locate the markers.

When we combined the two parts of the model, we found that it was not possible to create good reconstructions of the tissue attenuation at the same time as locating the markers. The sampler became stuck in a local mode: where there were really markers, the tissue attenuation value was increased; where the modelled

markers were, the tissue attenuation decreased to create a 'black hole'. In this way, the total attenuation matched the measured data.

We were able to use both parts of the model with a high value of the tissue attenuation smoothing parameter β. The smoothing was so aggressive that no structure in the tissue attenuation was recovered, but we obtained accurate estimates of the marker location.

We then demonstrated the application of our 3D model to a patient data set, with similar conclusions. The model including tissue attenuation only was able to recover structure, with fewer artefacts than the FDK reconstruction.

When we tried to replicate the phantom result where we accurately located the markers by using aggressive smoothing of the tissue attenuation, we found that the highest posterior probability of the marker location corresponded with artefacts in the FDK reconstruction which were used to estimate the integral of the tissue attenuation, $g_{\text{tissue}}(x; u, v, \theta)$. This issue persisted, even when we used a largely marker- and artefact-free reconstruction obtained using the analysis presented in Chapter 5.

Using a different method (based on the data) to estimate $g_{\text{tissue}}(x; u, v, \theta)$, we were able to accurately locate the markers. Estimating the location of the markers is our primary objective; once we have done this, we can locate the markers in the 2D projections, remove them from the projections and create a marker- and artefact-free reconstruction (our secondary objective).

The tissue attenuation was included in the model for two reasons. Firstly, successfully reconstructing the tissue attenuation values in the case where we are also estimating the marker location would give us an artefact-free reconstruction. We can achieve this in a different way (by processing the projection images, having located the markers).

Secondly, and more importantly, we need to know the 'background' value in the projections, $g_{\text{tissue}}(x; u, v, \theta)$, to which we add the value of $g_{\text{marker}}(x; u, v, \theta)$, in order to accurately locate the markers. We have demonstrated that, for this data set, using a data-dependent value which is constant for each projection is good enough to get reasonably accurate estimates. We shall suggest other ways that we could calculate an estimate for $g_{\text{tissue}}(x; u, v, \theta)$ in Chapter 7.

6.7 Appendix: extension to other marker shapes

The analysis presented in this chapter has used a very simple template for the markers—two straight lines. However, the implementation has been designed so that it would be easy to use a different template. We require the following functions to be defined:

- $M_i(t; x_m) : \{[0,1], \mathcal{X}\} \to \mathbb{R}^3$, defining the 'spine' of marker i. For the straight markers, this was the central axis of the cylinder.
- $T(t; x_m) : \{[0,1], \mathcal{X}\} \to \mathbb{R}^3$, the tangent to M_i.
- $\pi(x_m)$, the prior probability of the marker.

For the straight markers, we can write the set of points in marker i as $M_i = M_i^c \times \mathcal{B}_2(0.375)$, where $\mathcal{B}_2(0.375)$ is the disc of radius 0.375mm perpendicular to M_i^c, the central axis joining the two end points. We then defined the additive contribution to the mean pixel value, $g_{\text{marker}}(x_{\text{marker}}; u, v, \theta)$, to be proportional to the volume of the intersection of M_i and the pyramid with the pixel at its base and the X-ray source at its apex. We estimated this volume by sampling points $(\mathfrak{h}, \mathfrak{j}, \mathfrak{k}) \in M_i$. We then used the projection operator to calculate the corresponding point in the projection at angle θ, that is $\mathcal{P}_\theta((\mathfrak{h}, \mathfrak{j}, \mathfrak{k}))$. The contribution to the mean pixel value for pixel $p(u, v)$ in projection θ is proportional to the number of points sampled in M_i such that $\mathcal{P}_\theta((\mathfrak{h}, \mathfrak{j}, \mathfrak{k})) \in p(u, v)$.

For marker shapes other than a straight line, we suggest a slightly different definition for $g_{\text{marker}}(x_{\text{marker}}; u, v, \theta)$. Recall that the markers are, in fact, made of a coil of thin gold wire. Where the marker is bent, we expect to have less attenuating material on the outside of the curve. We therefore suggest a definition which considers the markers to have non-uniform 'density'. We define the contribution as

$$g_{\text{marker}}(x_{\text{marker}}; u, v, \theta) \propto M_i \times \mathcal{B}_2(0.375) \cap \Delta(u, v, \theta).$$

That is, the marker consists of those points lying on discs of radius 0.375mm whose centres lie on the spine of the marker. However, the marker density is not equal over its entire volume. Consider infinitesimal volumes within the marker,

on the inside and outside of a curve. The volume on the inside of the curve will intersect the discs whose centres are in a certain interval on the marker spine; the corresponding interval for the volume on the outside of the curve will be smaller. The contribution to $g_{\text{marker}}(\cdot)$ is proportional (as before) to the intersection of this non-uniform marker and the pyramid between the source and pixel.

This definition of $g_{\text{marker}}(\cdot)$ would allow it to be estimated in the same way as it was for the straight line:

1. Calculate a set of points $\{a_j\}$ uniformly spaced along M_i^c.
2. For each j, calculate the tangent to the spine, $T(a_j)$.
3. For each j, generate a point b_j uniformly in the disc of radius 0.375mm in the plane perpendicular to $T(a_j)$ whose centre is at a_j.
4. The estimate $\hat{g}_{\text{marker}}(x_{\text{marker}}; u, v, \theta)$ is then proportional to the number of points b_j which are projected into pixel (u, v) in projection θ.

CHAPTER 7

CONCLUSIONS

In this thesis, we have developed various statistical approaches to solving two problems in cone-beam computed tomography (CBCT): using X-ray projection data to estimate the locations of the ends of fiducial markers in 3D space, and creating artefact-free reconstructions of the patient. In this concluding chapter, we shall review each of the methods presented, discussing their strengths and weaknesses and further work which may overcome some of these, and suggest ways in which the analyses might be used in a clinical environment.

7.1 Statistical analysis as a pre-processing step

The first approach (presented in Chapter 4) used a model for the pixel data in each 2D projection. We generated estimates of the true pixel values. The prior probability model favours realisations which are locally smooth, using a clique potential chosen for its edge-preserving properties. The use of the marginal posterior mean (MPM) image acts as an edge-preserving smoother. When these MPM images are used in a conventional Feldkamp-Davis-Kress (FDK) reconstruction, the reconstruction appears visually superior to one carried out using projection images to which a Hamming filter has been applied.

Our model in this chapter did not explicitly include the markers, and so did not address either of the problems of locating the markers or creating artefact-free reconstructions. The use of a Markov chain Monte Carlo (MCMC) sampler is quite computationally intensive when the only desired outcome is a smooth image where edges are conserved. Other non-linear filters may produce similar results with less computational effort. However, the work in this chapter demonstrates the concepts and challenges which are addressed in the next two.

7.2 Modelling the markers in the 2D projections

Our first attempt to explicitly model the markers was presented in Chapter 5. We used morphological analysis methods to generates regions of interest (ROIs) in each of the 2D projection images. We then defined a pixel-based model describing the appearance of the markers in these ROIs. We used an MCMC sampler to find realisations of the pixellised marker which have a high probability in the posterior distribution. For each projection image, we combined the results from all the ROIs in that image, resulting in a probability assigned to each pixel in the projection image, representing the posterior probability that the end of a marker is projected into this pixel.

The results for each projection image are not perfect. In some cases, no ROI containing a particular marker is found in the morphological analysis; in other cases both markers appear in one ROI. Additionally, the sampler does not perform perfectly, sometimes becoming stuck in local modes. However, by combining the results from many projections, we may create a 3D voxellised map, where each voxel value represents an estimate of the probability that the end of a marker is contained in this voxel. This analysis results in accurate estimates of the location of the ends of the markers. However, we have not been successful in generating credible intervals for the marker location—the generated intervals were fractions of a millimetre wide.

We have also demonstrated a way of processing the projection images to remove the markers. When these processed images are used in an FDK reconstruction, the markers are largely absent from the reconstruction, along with the artefacts they cause.

Alternatively, the original and processed images may be combined to create a new processed image in which the appearance of the markers is reduced to a greater or lesser extent. Carrying out an FDK reconstruction using these images gives a reconstruction where the markers can still be seen but the artefacts are removed. This approach might be more acceptable to a clinician wary of over-reliance on image processing methods—the marker end location could be superimposed on slices from the reconstruction, and the success of the process verified.

7.2.1 Further work

This analysis uses a model of the projected marker in 2D space, and has produced accurate estimates of 3D location. Without using a 3D model (as we did in Chapter 6), it is not possible to exploit all the information in the images. However, there are some ways in which the use of the 2D model could be improved, which we shall now describe.

The model for the marker in the ROI is conditional on there being exactly one marker in the ROI. This is not always the case. Additionally, markers in some projections are not included in an ROI. An improvement to the model would be to allow two markers in an ROI. This could be incorporated into the analysis in the following way:

1. Carry out the analysis as described in Chapter 5, generating estimates of the locations of the ends of the markers in 3D space.

2. Using the projection operator $\mathcal{P}_\theta$, determine where the marker ends are projected in each of the projection images.

3. Generate ROIs which include these projected end locations, with sufficiently generous margins to allow for bent markers.

4. (a) For images where the two markers are projected into different ROIs, sample from the posterior distribution as before (i.e. continue to use the model which is conditional on there being one marker in the ROI).

(b) For images where both markers are projected into the same ROI, sample from the new model with two markers (i.e. use a slightly different model which is conditional on there being two markers in the ROI).

5. As before, calculate the probability that each pixel in each projection image contains the end of a projected marker, and combine these to form new estimates for the 3D location of the marker ends.

Carrying out this suggested procedure would particularly improve the way that information from two subsets of projections are used. These subsets are:

- Projections where the morphological analysis fails to find ROIs including the markers. These are usually lateral projections, where the markers are very unclear. With our current analysis, these projections do not contribute to the estimates of the marker location or the uncertainty. With the marker included in an ROI, they will begin to contribute.

- Projections where there are two markers in one ROI. Using a model which is conditional on there being one marker in the ROI usually results in the most probable projected markers being projections of the longer marker. The MCMC sampler tends to get stuck in the mode representing these markers, and so no information is gleaned about the shorter marker.

The estimates of the end locations are already accurate, though carrying out this procedure might improve their accuracy even further as information from a greater number of projections is effectively used. The main likely improvement will be in the creation of artefact-free reconstructions. The projections where both markers are not effectively removed in the processing step are mostly those where there is no ROI including the markers, or where the ROI includes two markers. By including each marker in an ROI and sampling from a model conditional on the correct number of markers, we predict that we would be successful in removing all but a few markers from the processed images. This would lead to reconstructions with a further reduction in artefacts.

7.3 Modelling the markers in 3D

The modelling approach in Chapter 6 is much more natural, with the markers parameterised in 3D space. Using data from a mathematical phantom, we demonstrated that we could sample from the posterior distribution of the markers to give an accurate estimate of the location, with an uncertainty of a few millimetres (more plausible that that found in Chapter 5). We also described a model for the attenuation of the patient's tissues, with which we obtained a good reconstruction of both the phantom and the patient. When we created reconstructions using the tissue attenuation model applied to patient data containing markers, there were fewer artefacts than in the FDK reconstruction.

For practical computational reasons, we were only able to use the tissue attenuation model on a small volume of interest (VOI) within the patient. Apart from restricting our search for the markers to this volume, this meant that we needed to use an alternative reconstruction to represent the patient outside the VOI. The FDK reconstructions we have been working with contain artefacts, which prevented us from identifying the correct location of the markers at the same time as estimating the tissue attenuation. For the 'straight markers' data set, we were however able to use the marker model in isolation to generate accurate estimates of the marker location.

7.3.1 Further work

We described in Chapter 6 how the model would very naturally extend to more complex 3D shapes. We required a curve in 3D to be parameterised by a function $M_i(t; x_m) : \{[0, 1], \mathcal{X}\} \to \mathbb{R}^3$, with the tangent vector defined for each t, and a prior probability defined for X_m. Possible classes of functions would be polynomials, finite trigonometric series, or other geometric curves such as segments of an ellipse.

In the phantom experiments, we discovered that we were not able to use our sampler to find the marker locations at the same time as recovering the structure in the tissue attenuation values. If we used a value of the 'smoothing' parameter β which was low enough to allow realisations with some structure to have reasonable probability, the sampler got stuck in a local mode where a 'black

hole' was created in the tissue attenuation values. We believe that the global mode is the true configuration, and one direction of further work would be to investigate different sampling approaches (such as tempering) to establish whether the posterior distribution could be analysed more effectively.

We need to estimate values of $g_{\text{tissue}}(x; u, v, \theta)$, to which we add $g_{\text{marker}}(x; u, v, \theta)$ to obtain the mean value for the normal likelihood. We used FDK reconstructions as estimates of X_{tissue}, from which we calculated $g_{\text{tissue}}(x; u, v, \theta)$. For the patient data, this resulted in the estimates of marker location being inaccurate—the most probable location for the marker was one in which the marker compensated for artefacts in the FDK reconstruction. We found that, for one particular patient data set, a data-dependent estimate of $g_{\text{marker}}(x; u, v, \theta)$, which was constant for each projection, was sufficient to generate reasonably accurate estimates of the marker location.

We now suggest alternative strategies for estimating $g_{\text{tissue}}(x; u, v, \theta)$. If we had sufficient computational resource, we could use a VOI which includes the entire pelvis (requiring either a lot of RAM for storing pre-calculated values or a lot of processor time for ray-tracing). Another solution might be to move the VOI across the whole pelvis, running a number of iterations at each location before moving it to another location, and repeating this process to sample from the posterior distribution for the whole pelvis. Our VOI is sufficiently big in the direction parallel to the axis of rotation (we only need the part including the markers), but we would need approximately 300 VOI locations to cover the transaxial plane (or more if we wanted them to overlap). The current analysis takes approximately one hour to run on a 2.7GHz laptop; simple arithmetic suggests that running the same number of iterations on 300 VOIs would take 12 days.

Another solution, which would be less computationally demanding, would be to consider the tissue attenuation to be constant within larger voxels. Sampling the posterior distribution for the tissue attenuation values for the whole pelvis would be possible if the voxels were large enough. In fact, we only need a few tens of millimetres in the direction parallel to the axis of rotation; we could include the entire transaxial slice at this thickness using voxels of pitch 4mm or 5mm with little more computational resource than we are currently using for our VOI.

Having created a low resolution MPM reconstruction of the whole pelvis, we could choose a small VOI in which to sample at high resolution. If we could choose that VOI in such a way that the only tissue in it was the prostate, we could reasonably assume that the values of X_{tissue} were constant. This would avoid the necessity of setting a too-high value of the parameter β—in our choice of VOI, we could create a situation where a very high value for β is, in fact, the correct choice.

Successful application of the tissue attenuation model to the whole pelvis would result in an artefact-free reconstruction. Another way to achieve this using the 3D model would be to process the 2D projection images, as we described in Chapter 5. In that chapter, we used the projection operator on the end points, and relied on the results of our 2D model to find the rest of the marker so that we could remove it from each projection image. With the 3D model, we know the location of the whole marker in 3D space, rather than just the ends. We can therefore project a number of points (possibly by using the stratified sampling approach we used in the MCMC analysis of this model), giving us a much more direct evaluation of which pixels in each projection contain the marker. We could then employ the same morphological analysis to create marker-free images which could be used to create an artefact-free FDK reconstruction.

7.4 Discussion

In this thesis, we have demonstrated that statistical models may be used to add to the information available from CBCT image data. We have given illustrative results, usually using one data set for each method. Once the improvements to the analyses described above have been made, the next task would be to make the methods generally applicable—for example, the parameters in the models have in general been empirically determined from a limited amount of data and require optimisation.

Having refined the models, they could be applied to a larger number of patients. There are different ways in which these analyses might be of use.

Firstly, they could be used on an individual patient basis. The projection images are acquired immediately prior to treatment. Currently, they are used in an FDK reconstruction. This 3D recontruction is registered to the planning CT data, and

the error in the patient's position calculated. The registration and assessment of positioning error may be carried out before the treatment commences. However, the registration takes a few minutes, and waiting for the results has drawbacks. The treatment couch is designed with reproducibility of positioning and radiolucency in mind, rather than patient comfort; the linear accelerator is housed in a bunker (for radiation protection purposes) which is not a bright and breezy environment. The patient's organs may move, either involuntarily or (especially if the patient is not comfortable) voluntarily. Also, the resources (radiographers and linear accelerator) are not being used.

For these reasons, many radiotherapy centres carry out an off-line analysis. Exact protocols vary, but CBCT images are often acquired before every fraction in the first week (i.e. the first five fractions), and then weekly thereafter. After the first week, the positioning is assessed for systematic errors. The treatment typically takes four weeks (20 fractions); any systematic error over a certain threshold noted in the first week is corrected for in subsequent fractions by changing the position of the couch (by translation and in some cases by rotation). The weekly image acquisition then serves as a check that nothing has changed.

The day-to-day variability in patient positioning may also be assessed using the first week's CT images. If the variability is large, steps may be taken to improve reproducibility (external supports if the bony anatomy is not reproducible; measures such as the use of enemas if organ motion is the problem).

The analyses presented in this thesis could be used in this process. We have provided estimates of marker location, which could be used instead of the voxel-based registration. With off-line analysis, the computation time required for the sampling would not be important. We have also provided estimates of the uncertainty in this measurement, which could be used to inform the decision on whether or to to correct the positioning error (knowing that the positioning of the couch is not perfect, with a typical uncertainty of a few millimetres).

Rather than being used on an individual patient basis, the statistical analyses could be used to inform the image acquisition and decision protocol. With the ability to infer information about position (and uncertainty in position) for a limited number of patients, it would be possible carry out optimisation of the protocols. Parameters that could be investigated might include the number and

timing of scans, the number of images acquired for each scan, and the amount of X-ray radiation used (and hence the noise level) for each image. With this approach, one could set an acceptable level of positional uncertainty (possibly related to the accuracy of couch positioning). One could then determine an optimal image acquisition protocol, taking into account factors such as the additional radiation dose and the resources required for scanning. It would also be possible to assess the criteria used to decide whether to correct the patient's position, even if the statistical analysis is not used on each patient to provide the input for those criteria.

Finally, a statistical analysis could be used during image acquisition. This would require a much lower computation time, but would allow optimisation of image acquisition on a scan-by-scan basis. An acceptable positional uncertainty could be specified prior to image acquisition. After acquiring some initial data, the position and uncertainty could be determined. If the uncertainty in a particular direction were higher than the acceptable level, further images could be acquired to reduce the uncertainty.

7.5 Concluding remarks

The use of statistical models to describe image data was pioneered in the mid-1980s. As computational resources become ever-more available, these methods have started to be used in routine practice, particularly in reconstruction of tomographic data in applications such as emission tomography, and more recently in CT. In this thesis, we have addressed the particular problem of implanted markers, successfully using statistical methods to obtain additional information and to improve reconstructions.

BIBLIOGRAPHY

[1] J. Besag, *Spatial interaction and statistical-analysis of lattice systems*, Journal of the Royal Statistical Society Series B-Methodological, 36 (1974), pp. 192–236.

[2] ——, *On the statistical-analysis of dirty pictures*, Journal of the Royal Statistical Society Series B-Methodological, 48 (1986), pp. 259–302.

[3] T. Buzug, *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*, Springer-Verlag, Berlin, 2008.

[4] J. Crook, Y. Raymond, D. Salhani, H. Yang, and B. Esche, *Prostate motion during standard radiotherapy as assessed by fiducial markers*, Radiother. Oncol., 37 (1995), pp. 35–42.

[5] CRUK, *Cancerstats mortality – UK*, tech. rep., Cancer Research UK, 2007.

[6] ——, *Cancerstats incidence 2008 – UK*, tech. rep., Cancer Research UK, 2011.

[7] A. P. Dempster, N. M. Laird, and R. D. B., *Maximum likelihood from incomplete data via the EM algorithm*, Journal of the Royal Statistical Society Series B-Methodological, 39 (1977), pp. 1–38.

[8] E. Dougherty, *Hands-On Morphological Image Processing*, SPIE, Bellingham, Wash., 2003.

[9] I. Dryden and K. Mardia, *Statistical Shape Analysis*, John Wiley, Chichester, 1998.

[10] I. A. Elbakri and J. A. Fessler, *Statistical image reconstruction for polyenergetic X-ray computed tomography*, IEEE Transactions on Medical Imaging, 21 (2002), pp. 89–99.

[11] ——, *Segmentation-free statistical image reconstruction for polyenergetic x-ray computed tomography with experimental validation*, Physics in Medicine and Biology, 48 (2003), pp. 2453–2477.

[12] H. Erdogan and J. A. Fessler, *Ordered subsets algorithms for transmission tomography*, Physics in Medicine and Biology, 44 (1999), p. 2835–2851.

[13] L. A. Feldkamp, L. C. Davis, and J. W. Kress, *Practical cone-beam algorithm*, Journal of the Optical Society of America a-Optics Image Science and Vision, 1 (1984), pp. 612–619.

[14] S. Geman and D. Geman, *Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 6 (1984), pp. 721–741.

[15] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov chain Monte Carlo in practice*, Chapman and Hall, London, 1996.

[16] P. J. Green, *Bayesian reconstructions from emission tomography data using a modified EM algorithm*, IEEE transactions on Medical Imaging, 9 (1990), pp. 84–93.

[17] U. Grenander, Y. Chow, and D. M. Keenan, *Hands: a pattern theoretic study of biological shapes*, Springer Verlag, New York, 1991.

[18] G. Han, Z. Liang, and J. You, *A fast ray-tracing technique for TCT and ECT studies*, in IEEE Nuclear Science Symposium & Medical Imaging Conference, 1999, pp. 1515 –1518.

[19] W. Hastings, *Monte Carlo sampling methods using Markov chains and their applications*, Biometrika, 57 (1970), pp. 97–109.

[20] S. S. Hebert, T. J. Gopal, *Maximum-likelihood preprocessing for improved filtered back-projection reconstructions*, Journal of computer assisted tomography, 18 (1994), pp. 283–291.

[21] G. T. Herman, *Image Reconstruction From Projections: The Fundamentals of Computerized Tomography*, Academic Press, New York, 1980.

[22] J. H. Hubbell and S. M. Seltzer, *Tables of X-ray mass attenuation coefficients and mass energy-absorption coefficients from 1 keV to 20 MeV for elements Z = 1 to 92 and 48 additional substances of dosimetric interest - NISTIR 5632*, tech. rep., NIST, 1989.

[23] H. M. Hudson and R. S. Larkin, *Accelerated image reconstruction using ordered subsets of projection data*, IEEE Transactions on Medical Imaging, 13 (1994), pp. 601–609.

[24] J. S. Kole and F. J. Beekman, *Evaluation of the ordered subset convex algorithm for cone-beam CT*, Physics in medicine and biology, 50 (2005), pp. 613–623.

[25] V. Kolehmainen, A. Vanne, S. Siltanen, S. Jarvenpaa, J. P. Kaipio, M. Lassas, and M. Kalke, *Parallelized Bayesian inversion for three-dimensional dental X-ray imaging*, IEEE Transactions on Medical Imaging, 25 (2006), pp. 218–228.

[26] P. La Riviere, J. G. Bian, and P. A. Vargas, *Penalized-likelihood sinogram restoration for computed tomography*, IEEE Transactions on Medical Imaging, 25 (2006), pp. 1022–1036.

[27] J. A. Lange and J. A. Fessler, *Globally convergent algorithms for maximum* a posteriori *transmission tomography*, IEEE Transactions on Image Processing, 4 (1995), pp. 1430–1438.

[28] P. Mayles, A. Nahum, and J. Rosenwald, *Handbook of Radiotherapy Physics: Theory and Practice*, Taylor and Francis, New York, 2007.

[29] C. N. Mitchell and P. Spencer, *A three-dimensional time-dependent algorithm for ionospheric imaging using GPS*, Annals of Geophysics, (2003), pp. 687–696.

[30] D. Moseley, J. Siewerdsen, and D. Jaffray, *High-contrast object localization and removal in cone-beam CT*, Medical Imaging 2005: Physics of Medical Imaging, pts 1 and 2, 5745 (2005), pp. 40–50.

[31] F. Natterer, *The Mathematics of Computerized Tomography*, Wiley, Stuttgart, 1986.

[32] C. Robert and G. Casella, *Monte Carlo Statistical Methods*, Springer, New York, 1999.

[33] G. O. Roberts, A. Gelman, and W. R. Gilks, *Weak convergence and optimal scaling of random walk Metropolis algorithms*, Annals of Applied Probability, 7 (1997), pp. 110–120.

[34] L. A. Shepp and B. F. Logan, *The Fourier reconstruction of a head section*, IEEE Transactions on Nuclear Science, 21 (1974), pp. 21–43.

[35] L. A. Shepp and Y. Vardi, *Maximum likelihood reconstruction for emission tomography*, IEEE Transactions on Medical Imaging, 1 (1982), pp. 113–22.

[36] B. D. Smith, *Image-reconstruction from cone-beam projections - necessary and sufficient conditions and reconstruction methods*, IEEE Transactions on Medical Imaging, 4 (1985), pp. 14–25.

[37] J. B. Thibault, K. D. Sauer, C. A. Bouman, and J. Hsieh, *A three-dimensional statistical approach to improved image quality for multislice helical CT*, Medical Physics, 34 (2007), pp. 4526–4544.

[38] H. K. Tuy, *An inversion-formula for cone-beam reconstruction*, Siam Journal on Applied Mathematics, 43 (1983), pp. 546–552.

[39] S. Webb, *The Physics of Medical Imaging*, IOP Publishing, London, 1988.

[40] B. R. Whiting, P. Massoumzadeh, O. A. Earl, J. A. Sullivan, D. L. Snyder, and J. F. Williamson, *Properties of preprocessed sinogram data in X-ray computed tomography*, Medical Physics, 33 (2006), pp. 3290–3303.

[41] C. F. J. Wu, *On the convergence properties of the EM algorithm*, The Annals of Statistics, 11 (1983), pp. 95–103.

[42] Y. B. Zhang, L. F. Zhang, R. Zhu, A. K. Lee, M. Chambers, and L. Dong, *Reducing metal artifacts in cone-beam CT images by preprocessing projection data*, International Journal of Radiation Oncology Biology Physics, 67 (2007), pp. 924–932.